ATCH 6

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
19th June, 1997

3. REPORT TYPE AND DATES COVERED
Final Technical Report

4. TITLE AND SUBTITLE
Computational Modeling of Expanding Plasma Plumes in Space Using a PIC-DSMC Algorithm

5. FUNDING NUMBERS
F49620-95-1-0319

6. AUTHOR(S)
Daniel E. Hastings and David Oh

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
33-413, Massachusetts Institute of Technology
77 Massachusetts Avenue
Cambridge, MA 02139

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR-97
0337

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NA
110 Duncan Avenue, Suite B115
Bolling AFB, DC 20332-0001

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

A comprehensive computational model of the plume of a Hall Thruster is presented. Spacecraft-plume interaction issues are addressed by studying how a partially ionized plasma expands in three dimensions and interacts with the surfaces it encounters. A quantitative analysis of fundamental parameters is used to show that the plume is an unmagnetized, quasi-neutral plasma in which Charge Exchange collisions play a significant role. The electrons are shown to be isothermal and collisionless and the electron momentum equation is shown to reduce to the Boltzmann relationship. An axisymmetric computational model of the plume is described in detail. The model combines the Particle-in-Cell (PIC) and Direct Simulation Monte Carlo (DSMC) methods to create a quasi-neutral PIC-DSMC model. The resulting model is relatively fast and is used to model meter scale objects on workstation class computers. Comparisons are made to experimental data from multiple sources. The model is shown to have good agreement with macroscopic quantities such as current densities and sputtering rates but poor agreement with microscopic quantities like the ion distribution function. Disagreement is thought to be due to inadequicies in the plasma source model. Further results are presented showing the thruster operating in vacuum. "Rules of thumb" are recommended for scaling erosion rates.

14. SUBJECT TERMS

15. NUMBER OF PAGES
270

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

Final Techical report for F49620-95-1-0319

Daniel Hastings & David Oh

33-413, MIT

Dept. of Aeronautics and Astronautics

77 Massachusetts Ave

Cambridge MA 02139

Period: 1 Aug. 96- Sept 30 97

COMPUTATIONAL MODELING OF EXPANDING PLASMA PLUMES IN SPACE USING A PIC-DSMC ALGORITHM

by

David Y. Oh

Submitted to the Department of Aeronautics and Astronautics on October 28, 1996
in partial fulfillment of the requirements for the degree of Doctor of Science in Aeronautics and Astronautics

Abstract

Electric propulsion (EP) offers the potential for substantial fuel savings for stationkeeping and other applications. One system which has shown particular promise is the Stationary Plasma Thruster (SPT) or Hall Thruster. These devices have a near optimum specific impulse for stationkeeping applications. The SPT thruster releases a partially ionized plasma plume in which charged particles are accelerated to high velocities. Satellite designers are concerned that the plumes from Hall Thrusters may erode and contaminate sensitive surfaces and interfere with communications signals.

A comprehensive computational model of the plume of a Hall Thruster is presented. Spacecraft-plume interaction issues are addressed by studying how a partially ionized plasma expands in three dimensions and interacts with the surfaces it encounters. A quantitative analysis of fundamental parameters is used to show that the plume is an unmagnetized, quasi-neutral plasma in which Charge Exchange collisions play a significant role. The electrons are shown to be isothermal and collisionless and the electron momentum equation is shown to reduce to the Boltzmann relationship. An axisymmetric computational model of the plume is described in detail. The model combines the Particle-in-Cell (PIC) and Direct Simulation Monte Carlo (DSMC) methods to create a quasi-neutral PIC-DSMC model. The resulting model is relatively fast and is used to model meter scale objects on workstation class computers. Comparisons are made to experimental data from multiple sources. The model is shown to have good agreement with macroscopic quantities such as current densities and sputtering rates but poor agreement with microscopic quantities like the ion distribution function. Disagreement is thought to be due to inadequicies in the plasma source model. Further results are presented showing the thruster operating in vacuum. "Rules of thumb" are recommended for scaling erosion rates.

A three dimensional computational model of the Hall Thruster's plume region is described and used to simulate a thruster mounted on a geosynchronous communications satellite. The results show that plume induced erosion of interconnectors can be mitigated through the use of appropriate thruster orientations and duty cycles. Erosion of the coverglass is not considered because accurate sputtering coefficients are not available. It is concluded that although SPT Thrusters should be canted with respect to the array, erosion of solar array interconnectors can be controlled and poses no fundamental barrier to the use of SPT thrusters on satellites.

Thesis Supervisor: Daniel E. Hastings
Title: Professor of Aeronautics and Astronautics

Acknowledgments

It is traditional for a graduating Ph.D. student to write a long set of acknowledgments in which they attempt to thank every single person who has ever contributed to their work in any manner. I think of this as a fool's errand, fraught with risks, since it is impossible to list everyone and the individuals the author omits are inevitably the only ones who read the acknowledgments. Nevertheless, I am bound by tradition, so I can only apologize in advance to those I neglect to mention.

I must begin by thanking my wonderful wife Bryn, who was there from the very beginning of this work. I thank her for her love and support through the qualifiers, the generals, though the endless long nights of work, and finally though the writing of this endless document. Her love and care has made this all worth doing, and I can not thank her enough for that.

I also thank my advisor, Professor Daniel Hastings who has been my mentor and guide throughout my graduate work here at MIT. His guidance was invaluable for both my Doctoral and my Master's work and I thank him for that.

I would also like to thank the members of my thesis committee: Professor Kenneth Breuer, Professor Chathan Cooke, Professor Manuel Martinez-Sanchez and Professor Leon Trilling. I give a special thanks to Professor Manuel Martinez-Sanchez his many conversations and suggestions and for putting up with my interruptions of his work.

At heart, I afraid that I am a theoretician, so for their help obtaining experimental data I would like to thank Tom Randolph at Space Systems/Loral, David Manzella at NASA Lewis, and Dr. Alec Gallimore at the University of Michigan. Without their data, everything in this thesis would be, quite literally, academic.

For their extensive help in computational matters, both with this thesis and just keeping things running in this crazy lab, I would like to thank Bob Haimes, Steve Ellis, and my lovely wife Bryn. Bob in particular has very kindly supported my strange questions and requests for help on matters computational. I would also like to thank my UROP's, Abbe Cohen and Andrew Ellis, for their help developing visualization and grid generation software.

My thanks to my family for supporting me through this process, even if they don't really understand it. But thanks for more than that of course; for supporting me though 21 years of education and for always encouraging me to excel without pushing me too far. It's a delicate balance, and one which I am sure to try to emulate with my own children. I appreciate the lesson in how to do it.

And quite literally last but not least, I would like to thank the many graduate student members of CASL and the SSL (now SERC, formerly SPPL). And I really mean it when I say last *but not least*. In my nine years at MIT, one of the most important things that I have learned is that what truly sets MIT apart is not its curriculum or its organization, but the quality of its students. I have been continually astounded and humbled by the students I've met at MIT. They work hard, they play hard, and they are generally brilliant. Dealing with my peers has been a humbling, exhausting, and exhilarating experience. Though I am happy to leave after nine years, this is one part of MIT which I will certainly miss. So thanks to you all for your help, your encouragement, and for putting up with me in general.

This work was supported by the Air Force Office of Sponsored Research under contract number F49620-95-1-0319.

"The known is finite, the unknown infinite; intellectually we stand on an isle in the midst of an illimitable ocean of inexplicability. Our business in every generation is to reclaim a little more land, to add something to the extent and solidity of our possessions."

-Thomas Huxley, on the Reception of the *Origin of Species* (1887)

"Hey Doc, we better back up. We don't have enough road to get up to 88."
"Roads? Where we're going, we don't need... roads."

-Back to the Future

Luck? Physics isn't luck. Physics is fact.

-Dr. Who

Table of Contents

List of Figures

List of Tables

Nomenclature

Note: Units stated in the text may supersede units given in this section.

A = Thruster Exit Area (m^2)

B = Magnetic Field Strength (Gauss)

$\mathbf{E}$ = Electric Field

E_k = Kinetic Energy of Particle

E_r = Radial Electric Field

E_z = Axial Electric Field

I_{sp} = Specific Impulse (seconds)

M_f = Fuel Mass (kg)

M_o = Initial Spacecraft Mass (kg)

T_e = Electron Temperature (Electron Volts)

T_i = Ion Temperature (Electron Volts or Kelvin)

W_{ion} = Ion Macroparticle Weight (unitless)

$W_{neutral}$ = Neutral Macroparticle Weight (unitless)

Z_i = Ionization Number (unitless)

a_o = Speed of Sound (m/s)

b_0 = Distance of Closed Approach between Collision Partners (Impact Parameter)

$\bar{c}_e$ = Mean Electron Thermal Velocity (m/s)

c_r = Relative Speed between Collision Partners (m/s)

e = Elementary Charge = 1.6×10^{-19} C

h = Grid Spacing (unitless)

k = Boltzmann's Constant = 1.38×10^{-23} J/K

m_e = Electron mass = 9.11×10^{-31} kg.

m_i = Xe ion mass = 2.18×10^{-25} kg.

$\dot{m}$ = Anode Propellant Mass Flow Rate (mg/sec)

n = Generic Number Density (m^{-3})

n_e = Electron Number Density (m^{-3})

n_i = Ion Number Density (m^{-3})

n_o = Reference Charge Density (m^{-3})

n_{ref} = Reference Electron Number Density (m^{-3})

p = Pressure (Torr or Pascal's)

r = Radial Position (meters)

R_a = Insulator Ring Width (meters)

t_{global} = Global Time Count (unitless)

t_{local} = Local Time Count (unitless)

$\mathbf{u}$ = Velocity

$v_\perp$ = Velocity Perpendicular to B field

v_z = Axial Ion Velocity (m/s or normalized)

z = Axial Position (meters)

Δt = Computational Timestep (unitless)

Γ_i = Ion Flux (number/m^2/sec)

β = Random number between 0 and 1 chosen from uniform distribution

β_p = Ratio of Kinetic Pressure to Magnetic Pressure

λ_d = Debye Length (meters)

λ_{en} = Electron-Neutral Mean Free Path (meters)

λ_{ee} = Electron-Electron Mean Free Path (meters)

λ_{ei} = Electron-Ion Mean Free Path (meters)

ε_o = Permittivity Constant = 8.85 x 10^{-12} F/m

μ_0 = Permeability Constant = 1.26 x 10^{-6} H/m

φ = Electrical Potential (Volts)

ν = Generic Collision Frequency (sec^{-1})

η_i = Ionization Fraction

ρ_{ge} = Electron Gyro Radius (m)

ρ_{gi} = Ion Gyro Radius (m)

σ = Generic Collision Cross Section (m^2 or cm^2)

σCEX = Charge Exchange Collision Cross Section

σE = Elastic Collision Cross Section

$\bar{\sigma}_{ee}$ = Mean Electron-Self Collision Cross Section

$\bar{\sigma}_{ie}$ = Mean Ion-Electron Collision Cross Section

θ = Angular Position (radians)

χ = Particle Scattering Angle (radians)

$\ln \Lambda$ = Spitzer Logarithm

CDEA: Closed Drift Extended Acceleration Thruster

CEX: Charge Exchange

DSMC: Direct Simulation Monte Carlo Method

NIITP: Scientific Research Institute of Thermal Process (in Russia)

PIC: Particle in Cell Method

PIC-MCC: Particle in Cell Method with Monte Carlo Collisions

RPA: Retarding Potential Analyzer

SPT: Stationary Plasma Thruster

TAL: Thruster with Anode Layer

Chapter 1: Introduction

The laws of motion are universal, so accelerating a vehicle always involves transferring momentum from the vehicle to its surroundings. Accelerating an object in space is therefore a uniquely difficult engineering problem. Vehicles on the ground have the advantage of being able to interact with their surroundings by pushing on other objects (cars) or by collecting mass and accelerating it in a controlled manner (airplanes and ships). Conventional space vehicles, on the other hand, can neither push on other objects nor collect mass from their surroundings. Accelerating a spacecraft therefore requires either some form of massless momentum transfer or the use of on board mass to carry momentum. Several methods of massless momentum transfer have been proposed for specialized applications (tethers, lasers, solar sails), but most currently practical space missions require the use of thrusters to accelerate propellant away from the spacecraft in a controlled manner.

The amount of acceleration experienced by a spacecraft can be determined by setting the change in the spacecraft's momentum equal to the momentum carried by the exhaust. This results is the well known rocket equation.

$$1-\frac{M_f}{M_o}=\exp\left(-\Delta v/c\right) \qquad (1.0\text{-}1)$$

Where c is the exhaust velocity and Δv is the velocity change imparted to the vehicle. The spacecraft's fuel mass fraction depends exponentially on the exhaust velocity, so it is generally desirable to use as high an exhaust velocity as possible. This point can be illustrated using a simple example. When a satellite is placed in geosynchronous orbit, the interaction of the Earth, Moon, and Sun's gravitational fields cause the satellite to drift in longitude (i.e. in the North-South direction). Thrusters are used to provide Δv to compensate for this drift. A typical geostationary communications satellite (geo-comsat) weighs about 2000 kg. and requires a Δv of 51.4 m/s per year for North-South stationkeeping [*Larson and Wertz* pg. 151]. The total Δv required over the course of a 12 year lifetime is therefore a quite substantial: 617 m/s. A conventional satellite might use monopropellant hydrazine thrusters for stationkeeping. Monopropellant thrusters have an I_{sp} of about 225 seconds which corresponds to an exhaust velocity of 2200 m/s [*Larson and Wertz* pg. 644]. Using these values in equation (1.0-1) gives a fuel mass fraction of 0.24, so almost a quarter of the satellite's mass is consumed by fuel used for North-South stationkeeping. Additional fuel is required for East-West stationkeeping and for circularizing the satellite's orbit. A typical commercial satellite costs about $77,000/kg [*16.89 final report, Spring 1995*], so the payload mass displaced by fuel is nominally worth $37 million. This very high cost encourages spacecraft manufacturers to develop advanced thrusters with higher exhaust velocities.

The exhaust velocity of a conventional thruster is limited by the energy per unit mass released by some chemical reaction. The best liquid thrusters use cryogenic propellants and have a specific impulse of approximately 450 seconds. Achieving higher specific impulses requires the use of esoteric chemicals or external energy sources. Electric propulsion (EP) devices use electrical energy to accelerate a working fluid to very high exit velocities. Because the energy imparted to the propellant is limited only by the supply of electrical power, these devices can dramatically lower the amount of fuel that needs to be carried on the spacecraft. The SPT-100 thruster, for instance, has an effective specific impulse of 1600 seconds. If the previously described 2000 kg. geo-comsat were to use an SPT thruster for stationkeeping, only 3.9% of its mass would be consumed by fuel, thus saving almost 420 kg which is nominally worth $3.2 million in displaced payload mass. In reality, some of this savings will be consumed by the electrical hardware needed to support the thruster, but the potential savings is still quite large. After literally decades of development, a variety of EP devices are now being adapted for use on commercial geo-comsats. The Telstar 4 satellite, which was launched by Lockheed-Martin in 1993, uses arcjets for stationkeeping and Hughes Space and Communications recently won a contract involving the use of ion thrusters for stationkeeping. Additional thrusters of near term interest are the Pulsed Plasma Thruster (PPT) and the Field Emission Thruster, both of which may be suitable for use on a new class of small "microsatellites." Overall, the commercial world has shown a high level of interest in EP devices for use on spacecraft in the very near future. At the present time, however, only arcjet thrusters have been used as primary propulsion on a commercial space mission.

One device which has attracted particular interest is the Hall Thruster, also known as a closed drift thruster. The term "closed-drift thruster" refers to an open ended (gridless) EP device in which electrons are restricted by a magnetic field while ions are driven out the end of an open channel. Like ion thrusters, closed-drift thrusters work by ionizing a working fluid and then electrostatically accelerating ions across an electrical potential. In an ion thruster, however, the working fluid is ionized in a chamber and ions are extracted and accelerated by an electric field set up between two metal grids. The large potential gradient between the grids prevents electrons from entering this region, so an ion engine's thrust is limited by space charge effects. In a closed drift thruster, on the other hand, ionization and acceleration occur in an open channel in which ions and electrons freely intermix to form a quasi-neutral plasma. As a result, closed drift thrusters are not subject to space charge effects and produce more thrust per unit area than ion thrusters.

Hall Thrusters were first developed in the early 1960's when spacecraft designers were interested in EP because they believed that future space vehicles would be able to generate large amounts of power (perhaps 100 kW). Since the specific cost and specific weight of power was expected to drop dramatically in the 1970's, electric propulsion was seen as an economical near term alternative to chemical propulsion systems. For EP devices, there is a tradeoff between the thruster's specific impulse and the weight of the spacecraft power system. As the specific impulse increases, the weight of the propellant decreases. However, since the power required to run the thruster is proportional to the square of the exhaust velocity, a thruster with a high specific impulse also requires a larger power system. The optimum specific impulse is the one that minimizes the combined weight of the spacecraft's propellant and power systems. This optimum specific impulse increases as the specific power (the power to mass ratio) decreases. Early designers believed that the specific power of spacecraft would drop dramatically in the 1970's, so they emphasized the development of EP devices with relatively high I_{sp}'s from 5000-10000 seconds. This led decision makers to encourage the development of Ion thrusters over Hall thrusters [*Kaufman* 1983]. In addition, Hall thrusters are relatively unsteady devices exhibiting numerous plasma instabilities. Researchers believed that it would difficult to build a thruster without preventing these instabilities from occurring. For both these reasons, Western work on closed drift thrusters largely ceased in the early 1970's. However, early predictions of cheap and light space power sources proved optimistic, and ion thrusters were generally too inefficient for use at the power levels available to designers throughout the 1970's and 80's. In addition, spacecraft designers are generally quite conservative in the level of new technology they accept for their spacecraft. The use of EP was therefore limited to the laboratory and a few scientific test flights.

While Western scientists emphasized work on arcjets and ion thrusters, Soviet scientists continued to develop Hall Thrusters throughout the Cold War era. Operational flight hardware was developed in the 1970's and Hall Thrusters were flown on over 50 missions during the late 70's and early 80's [*Myers* 1993]. When the Soviet Union collapsed in the late 1980's, Russian companies began actively marketing Hall Thrusters for use on Western spacecraft. The device's high specific impulse and Russian flight heritage have generated a great deal of interest among Western satellite manufactures, and a group of companies including Space Systems/Loral, Atlantic Research, Fakel, and Aerospatiale have formed a consortium to market this device in Western countries. Space Systems/Loral plans to launch a commercial mission using the SPT-100 in the next two years. Like all EP devices, the success of the Hall Thruster will require controlling and lowering the perceived risk associated with these devices. Customers and decision makers demand a detailed understanding of the impact that any new technology will have on their spacecraft, and though Hall Thrusters are rapidly maturing, they are no exception to this rule.

One issue of overriding concern to decision makers is the interaction of the Hall Thruster's exhaust with surfaces of the satellite on which it is mounted. Like all thrusters, EP devices release a plume of material in the vicinity of the spacecraft. Much of this material moves away from the satellite and generates thrust, but some of it remains in the vicinity of the spacecraft and can affect the operation of the satellite. The issues surrounding conventional chemical plumes are fairly well understood at this time, but EP devices release a partially ionized plasma in which charged particles can undergo both short range collisions and long range Coulomb interactions. These plumes are not well understood and can adversely effect the operation of a satellite in a variety of ways including: [*Samanta Roy* 1995]

- Parasitic current drain to biased surfaces
- Degradation of solar array panels, thermal control surfaces, and optical windows due to effluent deposition
- Attenuation and refraction of electromagnetic wave transmission and reception due to enhanced plasma density
- Absolute and differential charging of the spacecraft
- Interference due to electrostatic and/or electromagnetic noise
- Surface phenomena such as spacecraft glow
- Modification of spacecraft radiation signature
- Alteration of electrical and thermal properties of surfaces

Understanding how the plume from an EP device expands and interacts with surfaces is a critical part of understanding how a Hall Thruster will impact the operation of a commercial satellite.

Although the importance of plume interaction issues has been recognized for some time, practical considerations make it difficult to study EP plumes in ground facilities. Very high vacuums are needed to operate thrusters on the ground and the vacuum tanks required often have a significant impact on experimental results. Those results which do exist have large uncertainties and may not apply to thrusters mounted on satellites and/or operating in the space environment.

The lack of reliable data makes it desirable to develop analytic and computational models of EP plumes and their interaction with solid surfaces.

This thesis addresses plume-spacecraft interaction issues by describing how a partially ionized plasma expands in three dimensions and interacts with the surfaces it encounters. Though the primary focus of this work is the plume from a Hall Thruster, the models developed can be applied to a variety of EP devices. The plume of a Hall Thruster is a timely subject because there is a high level of interest in these devices for commercial and military missions and because an extensive experimental database has been developed over the past five years. At the present time, there are no three dimensional models of the plume from a Hall Thruster. Those models which do exist are generally empirical and apply only to simple geometries. A comprehensive three dimensional particle-in-cell (PIC) plume model has been developed for ion thrusters, but this model does not apply to the relatively dense plasma present in the plume of a Hall Thruster for two reasons. First, because of the high plasma density, the ratio λ_d/L is much smaller in the plume of a Hall Thruster than it is in an ion thruster (see Chapter 2). As a result, modeling the plume with a traditional PIC model is beyond the capability of most computers. Second, the ratio of the neutral density to the plasma density (n_n/n_i) is of order unity in a Hall Thruster plume. The collision models used with PIC simulations presently require that $n_n/n_i >> 1$. Existing ion thruster plume models are too computationally intensive for most design applications.

This thesis describes a new computational algorithm which can simulate an extended range of plasma plumes including the plumes produced by ion and Hall Thrusters. The model combines two well known computational methods to produce a three dimensional simulation of a relatively dense plasma plume interacting with the surfaces of a spacecraft. A major advantage of the model is that it runs on workstation class machines, which means that it can serve as a prototype of a computational tool for detailed spacecraft design work.

This chapter presents a review of the experimental and theoretical work which has been conducted on the plumes from chemical and EP thrusters and their interaction with spacecraft. The first section begins with a brief discussion of previous work on plumes from chemical thrusters, and then gives and overview of the wide range of work that has been conducted on plumes from EP devices. The next section gives a detailed and comprehensive overview of the experimental and theoretical literature associated with plumes from closed drift devices. Finally, the last section describes several computational methods which have previously been used to model plumes and describes their major strengths and limitations. The chapter concludes with a description of the research approach and an outline of the rest of this thesis.

1.1 Review of Previous Work

The importance of plume-spacecraft interaction issues was first recognized in the early 1960's, and a variety of experimental and theoretical work has been conducted since that time. The vast majority of the work concerns plumes from chemical thrusters, and a brief survey of this work is presented in section 1.1.1 below. In addition, a variety of work has been conducted on plumes from EP devices and on generic methods for simulating plasmas and rarefied gases. Overviews of previous work on EP and on Hall Thrusters is presented in sections 1.1.2 and 1.1.3 respectively, and an overview of several different computational methods used to simulate plasmas and rarefied gases is given in section 1.1.4.

1.1.1 Overview of Chemical Thruster Plumes

The plumes of chemical thrusters operating in vacuum have been studied extensively for a variety of applications including stationkeeping, docking maneuvers, and lunar surface landings. What follows is a brief survey of previous work on chemical plumes based largely on *Dettleff* [1991] and *Samanta Roy* [1995]. The section is not intended to provide a comprehensive review, but an overview of a very well developed field of research. A more detailed review is available in *Roux et al.* [1984].

The plume from a chemical thruster operating in vacuum can be divided into three different regimes, as shown in Figure 1.1. The flow near the nozzle is a hypersonic flow in which the Knudsen number is relatively small. The gas in this region is often undergoing simple isentropic expansion (though embedded shocks may be present) and can be treated using continuum methods. As the flow expands, the density drops and the plume enters a transition region in which the mean free path is no longer small compared to the size of the plume. Eventually, the Knudsen number becomes large and the flow evolves into a free molecular flow in which molecules move independently of each other. The main difficulty in modeling the plume as a whole is managing the transition from continuum to free molecular flow. The earliest plume expansion models were constructed to study the impingement of the plume from the Apollo Lunar Excursion Module (LEM) as it landed on the surface of the moon [*Dettleff* 1991]. Because the LEM has a relatively large engine (producing 5000-30000 N of thrust), the boundary layers in the nozzle were small and the method of characteristics was used to model the flow both inside and outside the nozzle. This method was generally successful, though very computationally intense by the standards of the time [*Sibulkin et al.* 1963, *Hill et al.* 1966].

Later efforts were made to model the contamination and impingement of plumes from smaller thrusters used for applications such as stationkeeping. With these devices, the region of interest is often in the far field where free molecular flow occurs. In these cases, it is natural to treat the plume as an area in which streamlines are straight and the density falls as $1/r^2$. The result is a density distribution of the form

$$\rho(r,\theta) = \frac{A}{r^2} f(\theta) \tag{1.1-1}$$

where A and *f* are parameters of unspecified functional form. A variety of different constants have been proposed based on theory and empirical measurements [*Roberts et al.* 1964, *Albini* 1965, *Hill et al.* 1966, *Boynton* 1967], but a model by *Simons* [1972] has obtained widespread application in its original and extended forms. Simons derived formulas for A and *f* based on the thickness of the boundary layer within the nozzle. This was done based on the observation that the boundary layer has a significant impact on the plume's divergence angle. Different authors have used different methods to calculate the thickness of the boundary layer and have applied the Simons model effectively [*Lengrand* 1974, *Lengrand et al.* 1976]. Both the method of characteristics and the Simons model are invalid in highly rarefied regions of the plume. The onset of rarefaction can be expressed in terms of *Bird's* [1970] breakdown parameter

$$P = \frac{1}{\nu}\left|\frac{d(\ln \rho)}{dt}\right|$$

Where ρ is the density and ν is the collision frequency. As P becomes larger, the degree of rarefaction becomes higher and collisions have less influence on the flow field. *Bird* [1970, 1981] used DSMC calculations to show that the onset of transition region occurs when $P \approx 0.05$. One method which has been used to model the plume is to define a "freezing" surface of constant P at which the flow is assumed to suddenly transition from continuum to free molecular flow. This simplifies the plume model by ignoring processes occurring in the transition region. The only practical method available for directly modeling the transition region is the Direct Simulation Monte Carlo Method (DSMC), which is a particle based computational method designed for the simulation of rarefied neutral flows. This method has been successfully applied to chemical thrusters and produces good agreement with experimental data [*Boyd* 1988]. Recent models have used the DSMC approach to model chemical reactions occurring within the plume as well. Therefore, aside from some detailed issues related to boundary layers, the manner in which plumes from chemical thrusters expand into vacuum is now fairly well understood.

The problem of plume impingement on surfaces has also been studied extensively. Again, the methods which can be used to model plume impingement depend very much on local Knudsen number. Figure 1.2 shows some of the possible flow regimes. Because the plumes from EP devices contain charged particles, much of the surface interaction work conducted for neutral plumes is not directly relevant to this study. The reader is therefore referred to *Dettleff* [1991] for a review of the literature concerning neutral plume-surface interaction effects.

1.1.2 Overview of Electric Propulsion Plumes

While the modeling of plumes from neutral thrusters is a relatively mature field, the plumes of EP devices are not as well understood and the study of plasma plumes is an area of ongoing work. The presence of charged particles in the plume also leads to a different set of interactions with surfaces than is seen with traditional chemical thrusters. At the present time, three different EP devices are approaching commercial viability: arcjets, ion thrusters, and closed drift thrusters. This section provides an overview of plume interaction models and experiments conducted with arcjets and ion thrusters. It is a survey of recent work and is not intended to be comprehensive. A comprehensive review of work on plumes from Hall Thrusters and other closed drift devices is given in section 1.1.3 below.

Ion Thrusters:

Ion Thrusters are gridded EP devices in which ions are electrostatically accelerated across an electric field formed between two parallel grids. The ions are produced by creating a plasma in an ionization chamber and then extracting ions from the chamber. The electrons are extracted through a separate anode and reinjected by a neutralizer into the plume, so the thrust from these devices is limited by space charge effects. A typical ion thruster is shown in Figure 1.3. The first ion thruster was constructed in the late 1950's and ion thrusters have been a subject of great interest to Western researchers. "Flight-ready" models have been developed, and both NASA and the Hughes Space and Communications Company are planning on using ion thrusters on missions over the next couple of years. Over the past 30 years a variety of experiments have been conducted on various ion thrusters using a number of different propellants. Commonly used propellants include Mercury and Cesium, which were primarily used in the 1970's, and Xenon gas, which is used in most present designs. Recent experimental work includes work by *Monheiser* [1991], *Beattie and Matossian* [1992], *Pollard* [1994], *Takegahara et al.* [1993], *Fearn et al.* [1993] and *Crofton* [1995]. In addition to the many ground based experiments which have taken place, several flight tests have occurred with ion thrusters over the past two decades [*Cybulski, R.J. et al.* 1965, *Staskus et al.* 1970, *Worlock* 1975, *Olsen et al.*, 1978]. Excellent reviews of ion thruster plume work are available in *Carruth* [1982], *Byers* [1979], *Deininger* [1985], and *Samanta Roy* [1994].

Based on the work conducted over the past two decades, a standard model has emerged for the basic structure of the ion thruster plume. The plume is divided into four basic components: primary beam ions, Charge Exchange (CEX) propellant ions, neutral propellant efflux, and non-propellant efflux. The beam ions are the primary thrust producing components of the beam and consist of ionized propellant which has passed through the region between the grids. These ions typically have very high velocities (> 20,000 m/s) and move on nearly ballistic trajectories, largely unaffected by the relatively weak electric fields present in the plume region. Because the ionization process is less than 100% efficient, 10%-20% of the propellant entering the chamber leaves as neutrals which flow through the grids at their thermal velocity (~ 500 m/s). Because the neutrals move much more slowly than the beam ions, the neutral density is at least 100 times greater than the ion density throughout most of the plume [*Samanta Roy* 1994]. The neutral propellant efflux not only interacts with surfaces, but also interacts with beam ions through charge exchange collisions. A charge exchange collision occurs when a fast ion and a slow neutral trade an electron through a resonance process to form a fast neutral and a slow ion. The resulting slow ions are referred to as CEX propellant ions because they are created by CEX collisions. In order to fulfill the requirement of overall quasi-neutrality, the beam ions create a radial electric field which prevents electrons from leaving the main beam. Although the fast moving beam ions are largely unaffected by this field, the slow moving ions created by CEX collisions are accelerated sideways and backwards by the field and can end up impacting the surfaces of the spacecraft. Some authors further subdivide CEX ions into three groups: those which are created in the grid region and emitted at angles up to 90 degrees from the centerline ("Group II"), those which are produced downstream and expand out of the beam ("Group IV"), and those which are attracted back and impact the acceleration grids ("Group III") [*Staggs et al.* 1968]. Ions striking the grid cause sputtering to occur, which introduces a fourth type of material into the plume: non-propellant efflux. The non-propellant efflux consists of neutral particles produced by sputtering from the grids and from the walls of the discharge chamber. The grid material is typically molybdenum, a metal with a low coefficient of thermal expansion. The low thermal expansion coefficient is desirable to help maintain a constant spacing between the grids during thruster operation. Molybdenum can deposit onto solar array cover glasses and cause measurable power losses. To combat the grid erosion problem, researchers have developed prototype carbon-carbon grids for use with ion thrusters. These grids are much more resistant to sputtering and may help mitigate contamination problems.

Throughout the 1970's, mercury and cesium were the most commonly used propellants for ion thrusters. These propellants have the potential to condense and/or plate sensitive surfaces of the spacecraft and adversely affect its operation. Partially as a result of efforts to alleviate these concerns, Xenon became the propellant of choice during the mid-1980's. Xenon is a noble gas and is therefore unlikely to plate or otherwise interact with surfaces of the spacecraft. The primary issue of concern to designers now is deposition of metal from non-propellant efflux. This issue continues to be addressed though a variety of laboratory studies and theoretical models.

In addition to the standard model described above, several important observations have been made about the plume over the past 20 years. *Sellen et al.* [1965] were the first to experimentally verify that the plasma density, plasma potential, and electron temperature in the plasma beam obey the following relationship

$$n_e(r,z) = n_e(0,z)\exp\{-e[\phi(0,z) - \phi(r,z)]/kT_e\} \tag{1.1-2}$$

This relationship is often referred to as the "Barometric Law," though we refer to it as the Boltzmann relationship. It will be used repeatedly in this thesis, though generally in the form

$$n_e = n_o \exp\{-e\phi/kT_e\}$$

where n_0 is understood to be the electron density at the point where the potential is zero. *Kaufman* [1975] showed that the Boltzmann relationship holds qualitatively within the CEX plasma as well as within the beam itself. Quantitative measurements of CEX densities and velocities are complicated by the need to account for facility produced background plasma in measurements of the backflow region. Again, a variety of efforts have been made to determine the importance of facility effects on CEX measurements, and different studies report that facility ions have anywhere from an "insignificant" to an "overwhelming" effect on CEX plasma measurements [*Carruth* 1982]. In particular, Komatsu and Sellen measured the contribution from CEX and facility plasma and showed that CEX plasma dominates the exit plane, the facility plasma quickly comes to dominate measurements taken in the downstream region. The facility and thruster size are both thought to have a significant effect on laboratory measurements, and facility effects must be considered when evaluating ground based CEX plasma measurements.

A series of different models of the ion thruster plume have been developed with varying levels of complexity. Point source models similar to those used for chemical plumes were developed by *Kaufman* [1975] and *Carruth* [1981]. These models are relatively simple and capture little of the physics of the plume expansion. *Komatsu and Sellen* [1976] used a particle tracking model based on experimental measurements to show that CEX ions leave the main beam in a radial direction, but did not model the backflow region. *Robinson* [1982] used a particle tracking method to study CEX ions, but failed to achieve good agreement with experiment in the backflow region. *Katz* [1981] developed a three-dimensional isothermal fluid model of the ion thruster plume which included CEX ion effects. But since the CEX ion flow in the backflow region is very rarefied, it is unclear that a fluid formulation applies in that region. *Peng* [1991-1993] used a three dimensional hybrid PIC-MCC model to track CEX ions in the vicinity of the grids and model the sputtering of material from the grids. Finally, two comprehensive particle based computer simulations have been developed for the plume from an ion thruster. *Samanta Roy* [1995] used a hybrid PIC and neutral fluid model to simulate the plume in three dimensions on meter length scales. His model uses analytic models to simulate beam ions and propellant neutrals, but uses the hybrid PIC method to simulate CEX ions. His method compares well to experimental results, but requires a massively parallel computer to compute large scale problems. *Wang et al.* [1996] developed a three dimensional PIC-MCC model of the ion thruster plume. Their model uses particles to simulate all species in the plume of the NSTAR ion thruster. Efforts are now being made to implement this model on supercomputers so that large computational domains can be simulated. Overall, a variety of different models have been developed for ion thruster plumes with some success at duplicating experimental results. However, at the present time, none of the codes developed are appropriate for design work, as they are either insufficiently accurate or require large computational resources to model meter scale three dimensional domains.

Arcjets

At the present time, arcjets are arguably one of the simplest and most mature EP technologies. Arcjets are electrothermal thrusters that use an electrical arc to heat a working fluid which is then accelerated through a conventional expansion nozzle. The arc is set up between a pointed cathode and an annular anode as shown in Figure 1.4. The arc partially ionizes the working fluid as it passes through the device, though ionization fraction is small compared to that found in ion or closed-drift thrusters. Measurements in the plume region give ionization fractions less than 1% at a distance of 30 cm from the nozzle exit [*Carney* 1988]. Arcjets were first developed in the early 1950's and have matured to the point that arcjets are now used for stationkeeping on some communications satellites [*Ruyten* 1993]. The specific impulse of an arcjet is low compared to ion or Hall Thrusters, generally between 400 and 600 seconds with Hydrazine propellants though specific impulses as high as 110 seconds have been achieved with hydrogen arcjets. Their simple design and ability to use hydrazine as a working fluid has made them attractive for near term use. They also perform better than Ion and Hall thrusters in time constrained missions where relatively high thrust levels are desirable from an EP device.

As with ion thrusters, a series of different experimental measurements have been made of the arcjet's plume region. Recent studies include *Carney* [1988], *Bogorad et al.* [1992], *Hoskins et al.* [1992], *Liebeskind et al.* [1993], and *Ruyten et al.* [1993]. Further studies are listed in *Ruyten et al.* [1993]. *Carney* [1988] used a Langmuir probe to measure the electron number density in the plume and Liebeskind used Laser Inducted Fluorescence (LIF) to measure velocity and temperature profiles in the plume at distances up to half a meter from the nozzle exit. *Bogorad et al.* [1992] made an effort to examine experimentally the interaction between an arcjet plume and a highly charged surface. Their work is not very rigorous, but gives a qualitative indication that charged surfaces exposed to an arcjet plume reach ground potential within a few seconds. In addition, Bogorad found that the plume of their device did not cause significant surface degradation over 40 hours of exposure. Further testing is required to determine the impact of the plume after many hours of operation. A series of space based tests were conducted on a cesium arcjet flown by the USSR on Cosmos satellites launched around 1987 [*Borisov* 1993]. These satellites were flown in relatively low orbits where the Earth's magnetic field is thought to have a significant impact on the structure of the plume [*Borisov* 1991]. In these flight experiments, the plume was shown to have a significant impact on the communications downlink, creating a radio "shadow" in which no signals could be received. These effects were thought to be a result of interactions between the plume and the Earth's magnetic field, which created a "petal" like plume structure in the vicinity of the spacecraft. No radio interference has been reported from the arcjet which is presently operating on the western Telstar 4 satellite.

While a great deal of experimental work has been conducted on arcjets, relatively few efforts have been made to analytically or computationally model an arcjet's plume region. *Carney* [1988] proposed a point source model similar to Hill and Draper's variation of equation (1.1-1). Her model shows reasonable agreement with experimental data in some cases, but is less good in other cases and requires the use of empirical constants determined from laboratory data. *Ling et al.* [1991] combined a similar model with cold plasma and ray optics models to estimate the effect the plume will have on radio signals. His work indicates that the plume will generally have little effect on antenna performance. *Gabdullin et al.* [1993] constructed numerical solutions to the "eiconal" equations and used it to calculate the scattering of radio waves over several generic arcjet plume configurations. Neither the numerical method nor the underlying assumptions are described in detail. *Boyd* [1993] uses the DSMC method to model cold flow (no arc ignition) from an arcjet. He achieves reasonable agreement with experimental data but since the DSMC approach is designed for rarefied neutral gases, this approach can not be easily extended to operational arcjets. To date, no effort has been made to apply advanced computational techniques to model the plume of an operational arcjet and no comprehensive models exist for the plume of an arcjet thruster. Designers have chosen to fly these devices despite the lack of a comprehensive plume model.

1.1.3 Overview of Closed Drift Thruster Plumes

This section presents a comprehensive review of previous work on plumes from closed drift thrusters. The term "closed-drift thruster" refers to a class of open ended (gridless) thrusters in which electron motion is restricted by a magnetic field while ions are electrostatically accelerated out the end of a channel. The term Hall Thruster is also used to refer to closed-drift devices. Two different closed-drift thrusters are shown in Figure 1.5. Closed drift thrusters are typically axisymmetric devices with a magnetic field oriented in the radial direction and an electric field parallel to the thrust direction. The electron cyclotron radius is small by design, so the electron current has two distinct components: one which flows counter to the thrust direction and another which is perpendicular to both the ion current and the magnetic field. The latter component is commonly referred to as the "Hall" current, and is the source of the term "Hall" thruster.

There are two types of closed drift thrusters. The first is a device in which the anode is made of a dielectric material and is referred to as a Hall Thruster, Stationary Plasma Thruster (SPT) or closed-drift extended acceleration (CDEA) thruster. The nomenclature has become quite confused, as the term Hall Thruster is also used to refer to closed-drift thrusters in general and the term SPT thruster actually refers to a specific CDEA design. Though the term CDEA has fallen out of common usage, we use it here to avoid ambiguity. The SPT is a CDEA design manufactured by the Russian company Fakel. It is referred to by a number indicating the outside diameter of the acceleration channel in millimeters. Fakel has built a series of SPT thrusters including the SPT-70, SPT-100, SPT-140, and SPT-200 thrusters. The SPT-70 has substantial flight experience in the former Soviet Union and an SPT-100 is schedule to fly on an experimental French satellite in late 1996. The second type of closed drift thruster is the "space charge-sheath" thruster or "anode layer thruster." This is a device in which the anode is made of conductive material and the acceleration channel is relatively short, so ions undergo most of their acceleration in a narrow region in front of the anode. The TAL thruster is a type of Anode-Layer thruster and is manufactured by TsNIIMASh in Russia. It is also being marketed for use in the West, though it is not known to have been flown either in the West or in the former Soviet Union.

The CDEA and anode layer thrusters were both developed in the 1960's, but because designers were interested in thrusters with higher specific impulses and the SPT's unstable plasma was not well understood, most Western work on closed drift thrusters halted in the early 1970's [*Kaufman* 1983]. At the same time, researchers in the former Soviet Union continued to work on closed drift thrusters and created "flight qualified" designs which were used on over 50 Russian spacecraft during the 1980's [*Myers et al.* 1993]. Western experts were largely unaware of this work until 1991, when Russian companies began to market the SPT-70 and SPT-100 for use in the West. For this reason, relatively little published work on closed drift thrusters exists from the early 1970's through the end of the Cold War in 1991. Early Western work on closed drift thrusters took place before this period and has been summarized by *Kaufman* [1983]. This work is largely concerned with the mechanics of the thruster itself rather than the plume region. The review in this section primarily describes work conducted in the period from 1991 until the present. This work represents the bulk of Western experience with plumes from closed drift thrusters.

Experimental Plume Studies

A great variety of experimental work has been conducted on the closed drift thrusters of various types. Early experimental efforts on CDEA Thrusters were intended to help understand the physics of these devices and improve their thrust efficiency. A variety of measurements were made of conditions inside the acceleration channel including measurements by *Komurasaki et al.* [1991] and *Bishaev and Kim* [1978]. Much more work was conducted in the former Soviet Union, but much of it is not available in the West. In an effort to measure the divergence of ions in the acceleration channel, Bishaev and Kim used planar probes to measure the direction and magnitude of the ion current 0.4 mm from the channel exit [*Bishaev et al.* 1978 and *Gavryushin et al.* 1981]. Although the quality of the results is unclear (error bars are not provided), these results show that the ion beam has a high divergence angle and ions are leaving the anode at angles of up to 50 degrees from the centerline. Their work also showed that the gradient of the magnetic field has a substantial influence on the distribution of ions leaving the anode. Based on this observation, work was conducted to "tune" the magnetic field and increase thrust efficiencies. It is this effort which presumably resulted in the thrusters which are now referred to as the SPT-70 and SPT-100 thrusters.

Although Soviet researchers spent a great deal of time developing and constructing CDEA Thrusters, little of their work was published in the West until 1991. Then, suddenly, a wide variety of information was published on the operation and performance of the SPT thrusters, including measurements of their thrust, efficiency, specific impulse, current-voltage characteristics, and the ion current magnitude and ionization rate inside the acceleration channel [*Bugrova et al.* 1991]. As with most Soviet/Russian data of this type, the quality of these results is unclear. The methods used to collect the data are not described and error bars are not provided with these measurements. Though later work has verified many of the reported values (including the reported thrust, efficiency, and specific impulse), the experimental measurements taken inside the channel have not yet been duplicated and should be treated with some skepticism. The first published far field plume data was published by *Absalamov et al.* [1992]. These results include Retarding Potential Analyzer (RPA) measurements of the ion distribution and witness plate measurements of erosion rates on solar cell cover glasses one meter away from the anode exit. The experiments cover a limited region of the plume, from 0 to 67 degrees from the centerline, and measure the erosion rate over period of only 95 hours. Again, error bars are not presented and the erosion results are quite vague. In addition, some inconsistencies are present in the RPA data. These inconsistencies are discussed further in section 4.2.4.

When SPT thrusters became available to Western researchers, a variety of efforts were undertaken to duplicate and extend Russian experimental work. *Myers and Manzella* [1993] used a combination of probes to measure the electron density, electron temperature, and ion current density in the plume at distances of up to 4 m away from the anode exit. Their work is very thorough and error bars are provided throughout their data. The electron temperature is shown to be constant throughout much of the plume, and preliminary RPA measurements are presented showing that ions in the plume possess a wide range of energies that spans more than 100 Volts. *Manzella* [1993] measured the emission spectrum at the exit of an SPT-100 and made the only known attempt to measure the plume's composition. His results suggest that the plume is 1-5% Xe I (neutrals), 76%-89% Xe II (single ions) and 11.9%-19% Xe III (double ions). In addition, he noted that no measurable quantities of eroded material are present in the plume region. Manzella also documented the presence of "oscillatory" and "quiescent" thruster operation modes and differentiated them by examining the magnitude of oscillations in the discharge current. His results suggest that instabilities may play an important role in the operation of the SPT-100 thruster. Error bars were not given on any of these results. Later, *Manzella* [1994] used Laser Induced Fluorescence (LIF) to measure the velocity and temperature of ions in the plume region. He measured the axial and radial velocity of ions located 11 mm downstream of the anode exit and determined their temperature by examining the width of the emission spectrum. Again, error bars were not included. In 1995, Manzella used a planar probe to measure the ion current density in the plumes of an SPT-100, a T-100, and a D-55 thruster. The T-100 is a CDEA Thruster produced by NIITP in Russia and the D-55 is a TAL thruster produced by TsNIIMASh also in Russia. Manzella's measurements of the ion current density showed that the three plumes are very similar to each other. Manzella also demonstrated that varying the facility pressure has a substantial impact on the ion current density profiles, particularly in the backflow region. Again, although the work is thorough, error bars are not provided and some inconsistencies are present. In particular, the integrated beam currents measured in his work often exceed the thruster's discharge current, thus violating charge conservation laws. This discrepancy and steps taken to correct for it are described in detail in section 4.2.2.

One issue of paramount interest to spacecraft designers is the impact the plume has on the surfaces of solar arrays, including their cover glasses and interconnectors. Several experiments were conducted in which sample coverglasses and interconnectors were exposed to the plume for an extended period of time and then examined for erosion, contamination, and changes in optical characteristics. The first reported use of witness plates was by *Absalamov et al.* [1992], but the results presented are quite vague and their quality is unclear. *Pencil* [1993] conducted witness plate experiments on a recently manufactured SPT-100 in a vacuum tank at NASA Lewis and exposed pieces of Quartz to the plume of an SPT-100 for a period of 50 hours. The samples were placed 2 m from the anode exit at angles ranging from 0-90 degrees from the centerline. Some of these samples were mounted inside collimators. Collimators are cylindrical tubes with an open orifice at one end and a sample mounted at the other. The orifice limits the sample's effective field of view and helps prevent contamination due to material sputtered from the walls of the vacuum chamber. The collimators were oriented so the orifice faced the thruster's exit. Samples which were not protected by collimators showed net erosion near the centerline, but net deposition at angles greater than sixty degrees. The deposited material was, however, shown to be from the walls of the facility and not from the thruster itself. Samples protected by collimators showed no net erosion or deposition because the collimator orifices were too small to allow accurate measurements of plume interaction effects. The experiment was repeated with improved collimators by *Randolph and Pencil* [1994]. Both solar cell coverglasses and silver interconnectors were placed in collimators 1 m from the anode exit and exposed to the plume for 200 hours. The results include measurements of erosion rates, transmissivity, and absorbitivity. No deposition was measured on any of the samples. Pencil and Randolph also performed a witness plate test with an "end-of-life" SPT-100 thruster in 1996 [*Pencil and Randolph* 1996]. These tests were conducted because long duration tests had shown that the metal surfaces of the SPT-100 can become exposed by erosion after hundreds of hours of operations, introducing new materials into the plume region [*Garner et al.* 1995]. In these tests, coverglasses were again placed in collimators and exposed to the plume of an old SPT-100 thruster for 200 hours. The transmissivity and absorbitivity were measured as before at different angles with respect to the centerline. Visual examination of the samples showed that some coverglasses had experienced net deposition rather than erosion as before. The deposited material is assumed to be metal which was sputtered from exposed surfaces of the anode. Interestingly, net deposition only occurred in a fairly narrow range of angles from 70-85 degrees from the centerline. Samples placed at angles less than 70 degrees from the centerline experienced relatively high erosion rates due to the high plasma density at the core of the plume. Samples placed at angles greater than 85 degrees experienced very low deposition rates because they had no line of sight to exposed parts of the anode. In both cases, the net deposition rates turns out to be negative, and no deposition was seen on any of the samples in these regions.

The most recent studies of the plumes from SPT-100 and D-100 thrusters were conducted in a relatively new facility at the University of Michigan. Unlike the facility at NASA Lewis, the base pressure in the University of Michigan facility is quite close to operating pressure, so atmospheric contaminants may influence data taken in the facility [*Gallimore* 1996]. A wide variety of measurements have been made in this facility and the work is generally very thorough. Measurements taken of the SPT's include microwave measurements of the plasma density in the plume [*Ohler et al.* 1995], measurements of the phase shift and attenuation of L-band transmissions[*Ohler et al.* 1996], measurements of ion flux, neutral flux, heat flux, and the ion energy distribution within these plumes [*King* 1996], and very near-field measurements of the charge and ion current density [*Kim* 1996]. The near field and neutral flux measurements represent unique data which were not previously available for these devices. Additional experiments have also been conducted with SPT and TAL thrusters operating on Krypton propellant [*Ohler* 1995, *Marrese* 1995 and *Marrese* 1996]. These results are largely of academic value, as no plans presently exist to use Krypton as a propellant for an operational SPT thruster. However, these results do provide necessary data on the physics of the CDEA Thrusters and the details of their operation. They also provide another set of data which can be used to test proposed models of the plume region.

The SPT-100's discharge current profile is quite noisy and exhibits a variety of time dependent phenomena. Though the discharge current is nominally 4.3 Amps, current variations of 4 amps peak-to-peak have been observed in some modes of operation [*Manzella* 1993]. Similar though smaller variations have also been observed with the D-100 TAL thruster [*Marrese et al.* 1996]. In an effort to determine the influence of these instabilities on radio transmissions, *Dickens et al.* [1995] used a Langmuir probe to measure the time dependent variation in number density along the centerline. Density variations of up to 12% were recorded in the SPT-100 plume, while variations of only 3% were measured in the D-100 plume. It should be noted that the SPT-100 tested was an "end of life" model, and these results may not be representative of the beginning of life performance. Dickens also measured the time dependent phase shift of radio signals transmitted through the plumes of the SPT-100 and D-100 thrusters at a frequency of 6.2 GHz. Depending on the thruster and the geometry, the measured phase shift varied from 5 to 20 degrees. Based on these results, Dickens also constructed a simple plume model based on the model of *Carney* [1988]. This model is discussed in detail in the next section.

In summary, because Western companies have shown great interest in using the SPT and TAL thrusters, a large number of ground based tests have been conducted in order to characterize the plumes from these devices and measure their impact on exposed surfaces. Many aspects of the plume, particularly its charge and current densities, have been studied quite thoroughly. The SPT-100 in particular been the object of many experimental efforts. Nevertheless, the data available are occasionally contradictory and the magnitude of the experimental error is generally unclear. In addition, it is unclear how well these ground based test results apply to orbital conditions. No space based experimental data have been produced for either the D-100 or SPT-100 thrusters. An SPT thruster is manifested for a flight on a French satellite later in 1996. Until that flight occurs, the only data available are from ground based experiments.

Theoretical Work

Because relatively few data are available on processes occurring inside anode layer thrusters, little effort has been made to model these devices. In addition, although several models have been proposed to describe processes occurring inside the acceleration channel of CDEA Thrusters [*Fife* 1996, *Hirakawa* 1996], relatively few efforts have been made to model the physics of the plume region. Several simple models of the plume have been developed based on empirical relationships. *Dickens* [1995] constructed a model based closely on the model developed by *Carney* [1988] for arcjets. In his formulation, the number density in the plume is basically given by

$$n(r,\theta) = \frac{n_o r_o e^{-[\lambda(1-\cos(\theta))]^p}}{r^2 \cos^2\theta} \qquad (1.1\text{-}3)$$

where n_0 is the on-axis number density one meter from the anode exit and p and λ are empirical coefficients determined from laboratory data and r_0 is an arbitrary constant with units of length squared. Dickens' original formulation does not include the r_0, but it is required to make the units consistent. Dickens used a modulation term to model oscillations in density as spherical wavefronts emerging from the center of the thruster exit plane. The time dependent number density is then given by

$$n(r,\theta,t)=\frac{n_o r_o e^{-[\lambda(1-\cos(\theta))]^p}}{r^2\cos^2\theta}\left(1-m\left(\cos(f_m 2\pi t-rk_p)\right)\right) \qquad (1.1\text{-}4)$$

where m and f_m are additional empirical parameters. Steady state density profiles based on equation (1.1-4) match experimental data fairly well when m is between 0.5 and 0.7 and λ is between 40 to 60. However, predictions of the phase shift depend heavily on m and generally fail to match experimental values. The use of four empirical parameters limits the value of this model, particularly since it does not include the interaction between the thruster and ambient gases present in the test chamber.

Randolph and Pencil also constructed an empirical plume model in an effort to predict erosion and deposition rates on surfaces exposed to the plume of an SPT-100 [*Randolph et al.* 1994 and *Pencil et al.* 1996]. Their model uses an expression for the current density based on a two term Gaussian distribution with collisionless expansion, resulting in a density that falls as $1/r^2$. This expression was first proposed for use with CDEA Thrusters by *Ohler* [1995] and gives

$$j=\frac{R_o^2}{r^2}\left\{k_o\exp\left[-\frac{\sin^2\theta}{k_1}\right]-k_2\exp\left[-\frac{\theta^2}{k_3^2}\right]\right\}$$

k_1, k_2, k_3, and k_4 are empirical constants based on experimental data [*Randolph* 1996]. k_1 and k_3 are unitless constants and k_0 and k_2 have units of current density. A model for the sputtering coefficient of Xe on different surfaces was also developed which included the effects of energy and impact angle on the sputtering coefficient. The energy dependent sputtering yield was integrated over a range of experimentally determined ion energies to given the effective sputtering coefficient. The resulting model matches experiment fairly well when the ion impact angle is normal to the surface, but matches less well when the primary impact angle is 45 degrees. This makes the model useful for "rule-of-thumb" calculations, but again the presence of empirical parameters makes it difficult to extend results from laboratory to experimental conditions. Better models are clearly needed to fully characterize the plume region.

At the present time, only two efforts have been made to develop advanced models of a Hall Thruster plume. *Bishaev et al.* [1993] constructs a multi-fluid model of the plume based on kinetic descriptions of the ion and neutral distribution functions. Electrons are modeled as an adiabatic gas and charge exchange collisions are included using the Crook collision operator. The result is a complex system of equations which are solved on a Cartesian grid using an unspecified computational technique. The formulation is quite complex and the results appear to be limited to simple geometries. The plume is modeled in normalized coordinates to distances equivalent to half a meter from the exit of an SPT-100 thruster. A brief comparison is made between the model and experimental data and is shown in Figure 1.6. No units are included in the figure, and the quality of the comparison is unclear. The basic model appears sound, but the formulation quite complex and probably difficult to extend to realistic geometries.

Rhee and Lewis [1995] developed a hybrid Particle-in-Cell model of an SPT plume which combines an analytical model of the ion beam with a PIC model for the CEX ions. The method used is similar to the method employed by *Samanta Roy* [1995] to model plumes from ion thrusters. However, Rhee and Lewis use a modified PIC method which uses simulated particles to represent the difference between the ion and electron charge densities. It is our opinion that this method is fundamentally invalid. In plasmas, perturbations in charge propagate as waves, not as particles, and travel at the relevant wave propagation velocity. In the modified PIC method, charge perturbations travel as particles at a rate determined by the charge and mass of the particle. The author's claim that their method "by default, assumes that there is a neutralizing electron background," is also incorrect. PIC simulations must directly simulate electrons using either particles or a hybrid formulation. We therefore conclude that the method used is fundamentally flawed and the overall model invalid.

In summary, although a variety of different experimental studies have been undertaken of the plumes produced by CDEA and Anode Layer thrusters, relatively little effort has been made to model the CDEA Thruster's plume region. The models which do exist tend to be simple and empirical formulations which are difficult to apply to space operations. To date, only one valid effort has been made to develop a three dimensional model of the plume. This model uses a kinetic formulation to solve for the various species in the plume, and although the basic model is sound, the accuracy of the model is unclear. There is a fundamental need for an accurate, three dimensional model of a Hall Thruster plume which can be used to simulate realistic spacecraft geometries. The goal of this thesis is to describe the physics of expanding plasma plumes and develop a computational model which can simulate the plume from a Hall Thruster. The formulation used will be generally applicable to plumes from a variety of devices including ion thrusters, anode layer thrusters, SPT thrusters, and the quasi-neutral regions of plumes released by plasma contactors. This model will address a critical need within the spacecraft community by allowing designers to determine the impact that many types of plasma plumes will have on the operation of a spacecraft. The model will not include Coulomb collisions, so it will not apply to plumes in which the plasma density is much higher than that seen in the plume of a Hall Thruster. The model is also quasi-neutral by assumption, so it will not apply to plumes containing double layers or other non quasi-neutral regions.

1.1.4 Previously Developed Computational Methods

Plasma are used today in a variety of applications including electric propulsion, semiconductor processing, and fusion research. A substantial effort has also been made to model rarefied neutral gases both for space and terrestrial applications. In both cases, the traditional Euler and Navier-Stokes formulations do not apply because the flow can be highly non-Maxwellian and because they do not properly model transport properties in ionized gases. As a result, substantial efforts have been made to develop computational methods for simulating plasmas and rarefied gases. A variety of different formulations now exist, including the Molecular Dynamics method, the Particle-in-Cell method, Particle-in-Cell with Monte Carlo Collisions, and the Direct Simulation Monte Carlo method. Molecular Dynamics is sometimes used to study basic phenomena, but because the computation time required by the method scales as order N^2 or higher, it is not appropriate for large scale simulations. A detailed description of this method is given in *Lubachevsky* [1991]. This section reviews the most commonly used particle methods: the PIC, PIC-MCC, and DSMC algorithms. Because these methods are well established and used in a variety of applications, no effort is made to review the literature surrounding each technique. Instead, the basic techniques and assumptions used with each method are described in separate sections. It should be noted that the model used in this thesis is a combination of the PIC and DSMC methods and is subject to many of the limitations present in both methods.

Particle-in-Cell Method

The particle-in-cell method is used to model the motion of particles in self and externally induced conservative force fields. This technique has been applied to a variety of fields including astrophysics, thermodynamics, semiconductor device simulation and plasma physics. The PIC method is described in detail in books by *Eastwood* [1988] and *Birdsall and Langdon* [1991]. Fundamentally, the PIC method uses computational particles to model plasmas at a statistical and molecular level. A large number of computational particles are used to represent an even larger number of real particles moving in a conservative force field. In Cartesian coordinates, the basic equation of motion for a charged particle moving in an electric field is

$$\frac{d^2x}{dt^2}=\frac{qE_x}{m_i} \qquad \frac{d^2y}{dt^2}=\frac{qE_y}{m_i} \qquad \frac{d^2z}{dt^2}=\frac{qE_z}{m_i} \qquad (1.1\text{-}5)$$

Because the charge and mass of the charged particle always appear as a ratio in these equations, a single computational particle can be used to represent thousands of charged particles without changing the basic equations of motion. In fact, since the charge to mass ratio is independent of the number of particles, each computational particle can represent thousands of electrons or ions without affecting the fundamental physics of the simulation. The computational particles are generally referred to as "macroparticles" because they each represent a much larger number of real particles. The number of real molecules represented by each computational particle is referred to as the macroparticle weighting factor.

The basic idea behind the PIC method is to create a collection of macroparticles and boundary conditions which correspond to some physical system and then move the particles in a manner consistent with system (1.1-5). A flowchart diagram of the PIC method is shown in Figure 1.7. Over time, the system evolves like a real plasma and will reach a steady state solution. Since the PIC method is time accurate, it can also be used to study time dependent phenomena such as wave propagation and Landau damping. The main difficulty with this method is that solving system (1.5) requires a knowledge of the local electric field.. This electric field is induced by the particles themselves and therefore varies in time and space. From Coulomb's law, the electrical force between two charged particles is given by

$$F = \frac{1}{4\pi\varepsilon_0}\frac{q_1 q_2}{d^2}$$

Where q_1 and q_2 are the charges of the two particles and d is the distance between them. The total electrical force on a charged particle is the sum of the forces induced by every other particle in the plasma. Determining this force directly requires determining the distance between the particle and each of the other particles present in the simulation. Calculating the force is therefore an order N^2 operation (i.e. the computational time required scales as the square of the number of particles) and is computationally impractical for large numbers of particles. To avoid scaling problems, the PIC algorithm uses a different approach to determine the local electric field. The basic idea is to divide up the domain into cells and then determine the charge density by counting up the number of charges in each cell and dividing by its volume. Once the charge density is determined, the electrical potential can be determined by solving Poisson's equation.

$$\nabla^2\phi(x,y,z) = \frac{-\rho(x,y,z)}{\varepsilon_0}$$

Where $\phi(x,y,z)$ is the electrical potential and $\rho(x, y, z)$ is the charge density as a function of position. The local electric field can then be found by differentiating the potential.

$$\mathbf{E} = -\nabla\Phi$$

Weighting particles to a grid and differentiating the potential are both Order(N) processes. Solving Poisson's equation is typically an Order (M ln M) process where M is the number of nodes on the grid. Order(M) methods also exist, though the overhead associated with these methods make it more practical to use an M ln M method in many cases [*Press et al.*, pp. 827-888]. The net result is that the overall process of weighting particles, solving for the electric potential, and then differentiating to obtain the electric field is an O(N ln N) process. This allows one to model much larger domains than would be possible if one were to directly calculate the force on each charged particle.

The validity of the PIC method depends on having small enough cells to resolve all important physical processes and on having enough computational particles in each cell to obtain statistically valid samples. Empirical tests suggest that approximately 10 particles per cell are required to obtain accurate results using the PIC method [*Hockney* pg. 22]. The number of cells required to cover the domain determines the number of particles required for the simulation, which in turn determines the amount of memory and computational time required to obtain a solution. In a plasma, the fundamental length scale is the Debye length, which is given by

$$\lambda_{db} = \sqrt{\frac{kT_e\varepsilon_o}{ne^2}}$$

where n is the local plasma density. For the full PIC method to be accurate, the size of the computational mesh must be no larger than the Debye length, or the simulation will fail to resolve wavelengths of interest in the plasma. The Debye length is inversely proportional to the square root of the charge density, so it becomes very computationally intensive to model very dense plasmas on large length scales.

The computational particles are moved by integrating system (1.1-5) using a finite difference method. One commonly used method is the leapfrog method, which is second order accurate in time when applied to (1.1-5). Like the computational grid, the timestep which is used to advance the simulation is also limited by physical considerations. In general, the timestep must be small enough to resolve all frequencies of interest. In a plasma, the highest frequency of interest is usually the electron plasma frequency which is given by

$$\omega_{pe} = \sqrt{\frac{n_e e^2}{m_e\varepsilon_o}} \tag{1.1-6}$$

The minimum requirement to resolve this frequency is given by the following relationship [*Hockney et al.* pg. 335].

$$\omega_{pe}\Delta t < 2 \tag{1.1-7}$$

In practice, a smaller timestep is often required to resolve electron plasma wave phenomena adequately. Again, as the plasma density increases, the plasma frequency increases and more timesteps are required to reach the final solution. In addition, the need to meet the requirements of (1.1-7) make it very difficult to resolve physical phenomena occurring on ion plasma timescales. The ion plasma frequency, for instance, is given by

$$\omega_{pi} = \sqrt{\frac{n_i e^2}{m_i\varepsilon_o}}$$

In Xenon, m_i/m_e is 239,297, so ω_i and ω_e differ by a factor of 500. This difference makes it difficult to resolve ion plasma waves using a traditional PIC method. An alternative approach is to model the ions using a PIC method, but to model the electrons using a Vlasov or hydrodynamic formulation. This technique is known as a hybrid-PIC model. Because the electrons are treated separately from the ions, the highest frequency of interest in the PIC part of the simulation is the ion plasma frequency, so timesteps can occur on ion timescales. Another alternative to traditional PIC is to integrate the equations of motion using an implicit formulation which is stable even when $\omega_e\Delta t >> 2$. Such schemes have had some success though they can become very complex and are subject to their own timescale limitations [*Birdsall et al.* pg. 205].

In summary, the PIC algorithm is a well established computational method which has been used to simulate plasmas in a variety of different applications. Though more computationally expensive than traditional solvers based on the hydrodynamic equations, the PIC method is necessary when simulating highly non-Maxwellian plasmas.

Particle in Cell with Monte Carlo Collisions (PIC-MCC)

Although the PIC method is well suited to the simulation of collisionless plasmas, it can not be used to simulate collisional plasmas for two reasons. First, when a particle is weighted to the grid, the charge it carries is effectively smeared out in space, so particles in the same cell are unable to "see" and interact with each other. Second, since neutral atoms carry no charge, they do not contribute to the electric field in system (1.1-5) and can not influence the trajectories of other particles in the plasma. In many cases, the presence of neutrals can have a substantial impact on a partially ionized plasma. In order to model collisional phenomena, a modification of the PIC method known as the Particle-in-Cell method with Monte Carlo Collisions (PIC-MCC) has been developed. This model is described in detail by *Birdsall* [1991] and has been applied to low pressure, low density plasmas with "only a few particles colliding per field time step." The range of conditions in which PIC-MCC has been used is at pressures < a few 100 millitorr and plasma densities less than 10^{16} m^{-3} [*Birdsall* 1991]. Like the PIC method, the PIC-MCC method is based on the use of computational macroparticles to model plasmas statistically at a molecular level. In the PIC-MCC model, macroparticles are used to represent both ions and neutrals in the plasma. Just as in a conventional PIC model, particles are weighted to the grid, the electric field is calculated, and the particles are moved according to system (1.1-5) using finite-difference approximation. The neutral particles are not influenced by the electric field and do not contribute to the charge density, so they move on ballistic trajectories independently of the charged particles. The difference between the PIC and PIC-MCC methods is that an extra Monte Carlo Collision step occurs in each iteration, as shown in Figure 1.8. During the MCC step, the neutrals are weighted to the grid and the neutral density is calculated as a function of position in the plasma. Each charged particle is then assigned a probability that it will undergo a collision in this timestep given by [*Birdsall* 1991]

$$P_{collision} = 1 - \exp\left[-n_{neutral}\sigma_{total}(E_m)v_m\Delta t\right]$$

Where P is the probability that a collision occurs, $n_{neutral}$ is the local neutral density, $\sigma_{total}(E_m)$ is the energy dependent total collision cross section and v_m is the particle's velocity. The expression for P is valid when the total collision cross section is known and when the velocity of the charged particle is much greater than the velocity of the neutrals in the simulation. Once P is known, a random number β is chosen and compared to P. If $P > \beta$, the particle is scattered in this timestep. The velocities of scattered particles are then randomly modified to account for the dynamics of the collision. In CEX collisions, the particle is scattered isotropically and scattering angles are chosen from uniform distributions. At no time is a collision partner actually chosen for the scattered particle. This is a fundamental difference between the PIC-MCC method and fully collisional methods like DSMC. Since no collision partner is chosen, the relative velocity of the two particles is (strictly speaking) unknown and the effect of a collision on the second partner is not included. In CEX-like collisions (fast charged particle colliding with a slow neutral) the relative velocity depends almost solely on the ion velocity and the neutral velocity has little effect on the collision cross section or final ion velocity. In these cases, it is valid to model the collision without considering the influence of the second collision partner's velocity on the collision cross section. With collisions within a species, on the other hand, the relative velocities of the two particles depends strongly on both collision partners. Self collisions and collision between two species moving at the same velocity are therefore not accurately simulated by the PIC-MCC method. In addition, because the second partner is never chosen, the effect of CEX collisions on the neutral gas is *not* calculated, the velocity of neutral particles is not modified, and new neutrals are not produced by any of these collisions. Because these neutral effects are ignored, the PIC-MCC method is valid only when collisions can be said to have little effect on the neutral species' distribution function. This occurs when very few collisions occur each timestep or when the neutral density is much greater than the ion density (which is equivalent to requiring that the ion collision frequency be much higher than the neutral collision frequency). In principle, it may be possible to apply PIC-MCC to neutral particles in situations where the plasma velocity is much higher than the neutral velocity. In these cases, it may be possible to write a collision probability function for the neutrals based solely on the ion velocity. But because the PIC-MCC method models collisions based on the interaction between an individual particle and a bulk gas (as opposed to the interaction between two particles), it can never be applied in situations where self-collisions are important or where collisions have a substantial impact on the underlying distribution function of both gases. Many different forms of the PIC-MCC method have been developed and are described by *Birdsall* [1991].

In summary, the PIC-MCC method is a useful algorithm which has been used to simulate plumes from ion thrusters and other weakly ionized plasmas. The method is a modification of the basic PIC method and includes the effects of collisions between charged and neutral particles by modeling collisions between a fast particles and a slow bulk gas of known velocity and density. The method is computationally efficient, but does not involve the selection of collision pairs. For this reason, the PIC-MCC method is generally valid only when the neutral density is much greater than the ion density and in cases where the overall collision rate is too small to effect the underlying distribution functions. It should be noted that the PIC-MCC method has been used to produce a three dimensional particle simulation of the plumes from ion thrusters [*Wang et al.* 1995].

Direct Simulation Monte-Carlo Method

The DSMC method is a particle based computational method used to simulate rarefied neutral gases in which the mean free path is long enough that the fluid can not be treated as a continuum flow but short enough so that the gas can not be treated as a collisionless flow. Like the PIC and PIC-MCC methods, the DSMC method uses computational particles to simulate a gas statistically at a molecular level. Unlike the PIC and PIC-MCC methods, the DSMC method also uses statistical methods to model short range collisions between atoms and molecules. The modeling of short range collisions makes the method useful for the simulation of problems involving transition flow (as shown in Figure 1.1) and is described in detail in *Bird* [1994]. Electric fields are not considered in the DSMC algorithm, so this technique is not appropriate for the modeling of charged particles (though the DSMC method has been used to simulate the relatively collisional plasmas used in semiconductor processing). The DSMC method has been successfully used to model chemical reactions in rarefied gases for a variety of different applications.

The basic DSMC method is similar to the PIC and PIC-MCC algorithms. Computational macroparticles are used to simulate much larger collections of real particles moving in a computational domain. The initial and boundary conditions are chosen to match a physical problem, and the particles are moved according to Newton's laws until a steady state solution is reached. Like the PIC method, the DSMC method is a time accurate formulation and can be used to study transport phenomena, chemical reactions, and unsteady flow phenomena. An outline of the DSMC method is shown in Figure 1.9.

The most straightforward way to model collisions at a molecular level is to compare the trajectory of each molecule to the trajectory of every other molecule in the gas, determine when the next collision will occur, advance the simulation to that point, and then appropriately modify the two collision partners. This procedure is used in Molecular Dynamics simulations and is very computationally intensive as the computation time scales as square of the number of particles in the simulation. The DSMC method uses a different approach to the collision problem. The basic idea is to decouple collisions and move steps in the gas, so particles can be moved without considering all possible collision pairs. Collisions are then considered in a separate step. Though the motion of the individual particles will be incorrect (particles will appear to pass "through" each other without colliding at a microscopic level), the collective motion of many particles can modeled correctly over time by ensuring that the proper collisions occur between move steps.

The macroparticle themselves move according to system (1.1-5), but with the electric field set equal to zero. At the end of each move step, particles that end up in the same cell are grouped together using a fast sorting algorithm. Collisions are then carried out between randomly chosen pairs of particles within each cell grouping. The method used to ensure that the correct number of collision occur in the correct proportions is critical to the DSMC scheme and merits some discussion. A number of different statistical methods are used in various applications, but we will discuss one of the oldest and most common selection algorithms: the selection-rejection scheme with local time counters. A scheme will be described for simulations in which a single species can undergo a single type of collision.

As the name implies, the local time counter scheme requires that each cell have its own time counter. At the beginning of the simulation, both the local and global time counters are set equal to zero. At the beginning of each iteration, the global time counter in incremented by one time step. Then, after the particles have been sorted by cell, the simulation iterates through the cells and compares the value of a local time counter to the simulation's global time. If the local time is larger than the global time, the simulation advances to the next cell and no collisions take place. If the local time is less than the global time, the simulation randomly selects two potential collision partners from the particles present in the cell. The simulation then calculates their relative velocity and energy dependent collision cross section with no regard for the actual position of the particles within the cell. Since the position of the particles is ignored, another random selection must be made to determine whether or not the particles collide. In order to ensure that collision pairs are selected in the correct relative proportions, the selection must be weighted by the product of the particles' velocity and collision cross section. This is accomplished by selecting a random number and comparing it to the following ratio

$$\frac{\sigma c_r}{(\sigma c_r)_{\max}} \qquad (1.1\text{-}8)$$

The numerator is the product of the collision cross section and the relative speed of the two potential collision partners. The denominator is the largest value of this quantity previously seen in this cell. The value of $(\sigma c_r)_{\max}$ varies from cell to cell and is updated with each iteration. If β is greater than expression (1.1-8), two different potential collision partners are chosen and no collision occurs. If β is less than expression (1.1-8), a collision is assumed to occur and the two particles are modified accordingly. This selection scheme is referred to as the selection-rejection scheme and is appropriate when simulating a single type of collision. A modified scheme which can be used to simulate multiple collisions is discusses in section 3.5.1.

When a collision takes place, the local time counter is updated to reflect the fact that a collision has occurred. The amount by which the local time counter is advanced is also weighted by the collision cross section on the following basis. The average volumetric collision frequency between molecules in a gas is given by

$$\nu = \tfrac{1}{2} n^2 \sigma c_r$$

where v is the number of collisions per unit volume per second. This expression can be inverted to give a collision time. In the local time counter method, this quantity represents the time consumed by each collision, so the local time counter is advanced by the following quantity

$$\Delta t_{local} = \frac{2W}{Vn^2 \sigma c_r}$$

where V is the volume of the cell and W is weight of the macroparticles. This formulation is valid for a single species undergoing collisions with itself. Alternate formulations for multiple particles undergoing multiple collisions are discussed in Section 3.5.1. Once a collision occurs, the local time counter is again compared to the global time counter to determine if another collision is necessary. This process is continued until the local time in each cell is larger than the global time in the simulation, at which point another iteration can occur. Additional iterations take place until the flow reaches a steady state or the simulation is halted.

Since potential collision partners are selected randomly without regard for their location, one important limitation of the DSMC method is that the size of the grid cells must be smaller than a mean free path, or collisions will take place between particles that are physically too far apart for such collisions to take place. In addition, it has been empirically observed that statistically accurate results are obtained when there are more than ~20 macroparticles in each cell. This rule is independent of the collision cross section. As the mean free path gets smaller, the number of cells increases, which in turn increases the number of particles required. Overall, the number of particles required in a simulation increases with the number density, so very dense flows can not be simulated using the DSMC method.

The major advantage of the DSMC method is that the sorting, selection, and collision processes are all order N or order N ln N processes. As a result, the DSMC method can be used to simulate much larger problems than can be addressed by the MD method. To date, the DSMC method has been used for a wide range of applications and has been repeatedly verified through experimental studies. It has been an important tool for researchers exploring the physics of rarefied gases, including the expansion of neutral gas plumes into vacuum.

In summary, a variety of different computational algorithms exist which use computational macroparticles to statistically simulate gases and plasmas at a molecular level. These algorithms have been used to study plasmas and rarefied gases in a variety of different applications, but each method is subject to serious theoretical and practical limitations. The PIC method has been well established for the simulation of collisionless plasmas, but is fundamentally unsuited for the simulation of denser plasmas where collisions are a significant factor. The DSMC method is well suited to the simulation of rarefied gases, but does not include the long range interactions present in a PIC method. In addition, each method is subject to length and time scale limitations determined by the fundamental characteristics of the gases they simulate. The PIC-MCC method is a partial attempt to combine both short and long range interactions into single computational algorithm, but because it does not track the influence of collisions on neutral particles, it can only be applied when the neutral density is much higher than the charge density in the plasma. To date, no general algorithm has been constructed with the ability to fully model the interactions between a non-Maxwellian plasma and rarefied neutral gases.

1.2 Outline of Research

Although a great deal of work has been conducted on spacecraft-plume interaction issues, to date, no comprehensive 3D simulations have been constructed of the relatively high density plumes emitted by closed drift thrusters. This thesis will address this issue by examining the following fundamental questions:

- How does a partially ionized plasma plume expand into a vacuum?
- How does this plume interact with three dimensional solid surfaces it encounters?
- What computational techniques can be used to efficiently model a 3D plasma flow and its interaction with solid surfaces?

This thesis addresses these questions by describing a comprehensive computational model of the plume from a Hall Thruster. In particular, this work will consider the interaction between the plume from an SPT-100 and the surfaces of the spacecraft on which the thruster is mounted. Although the primary focus of this work will be on Hall Thrusters, the method is general and can be applied to plumes from ion thrusters, anode layer thrusters, and the quasi-neutral regions of plumes from plasma contactors. In principle, the model can be applied to all quasi-neutral, steady-state problems involving partially ionized collisional plasmas in which short-range coulomb collisions are unimportant. The quasi-neutral PIC-DSMC method can be applied to problems traditionally solved using the PIC-MCC method, though it may not be the most computationally efficient solution to the problem. It can also be used to solve problems which are not addressed by PIC-MCC such as plasmas in which neutral-neutral collisions play an important role or situations in which the neutral and plasma densities are of the same order of magnitude. The quasi-neutral PIC-DSMC model also requires much less computational work than previous models and can be applied to realistic spacecraft geometries. This makes the model a suitable prototype for the development of simulations to be used for spacecraft design work.

Chapter 2 describes the basic structure of the plume from a CDEA Thruster and uses some simple models to derive basic plume parameters. Chapter 3 gives a detailed description of an axisymmetric PIC-DSMC model of the plume and Chapter 4 presents results from the model as well as extensive comparisons to experimental data. Chapter 5 describes a three dimensional plume model based on the same fundamental algorithms and Chapter 6 presents simulations which verify the 3D model and show a CDEA Thruster mounted on a typical geosynchronous communications satellite. Finally, Chapter 7 presents conclusions and plans for future work on the PIC-DSMC plume model.

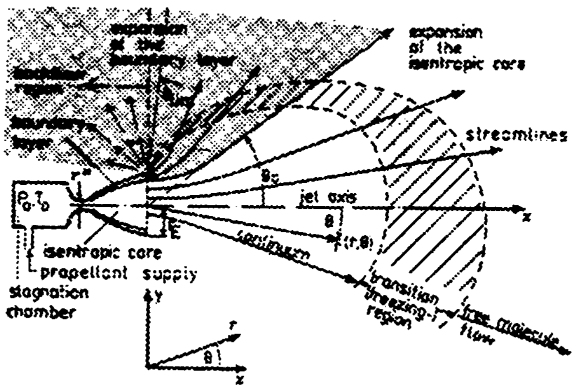


Figure 1.1: Flow Regimes Associated with a Plume Expanding into Vacuum
[Dettleff, 1991]

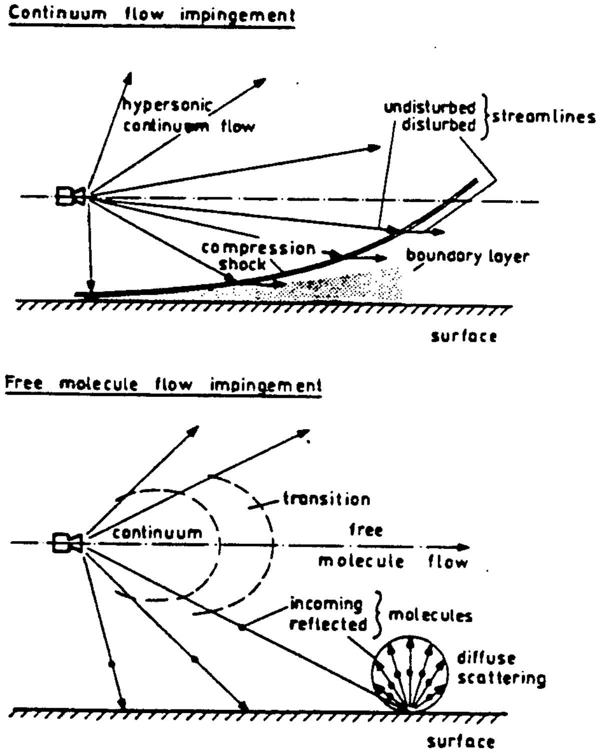


Figure 1.2: Regimes of Surface-Plume Interaction
[Dettleff, 1991]

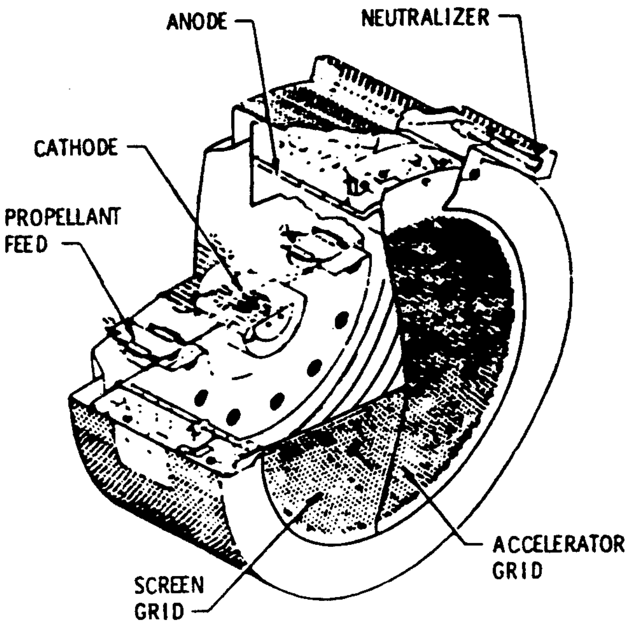


Figure 1.3: Picture of a Typical 30 cm Ion Engine

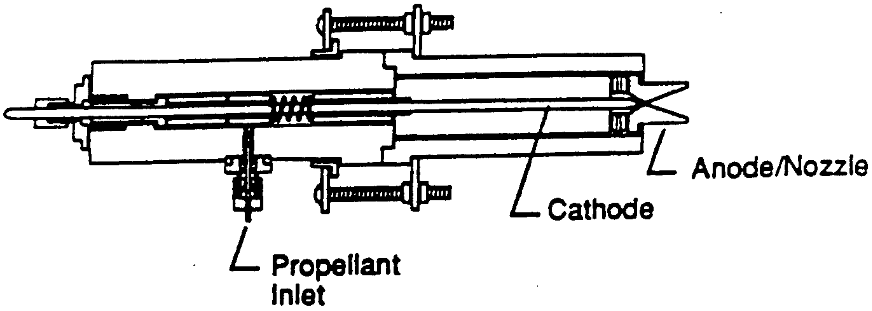
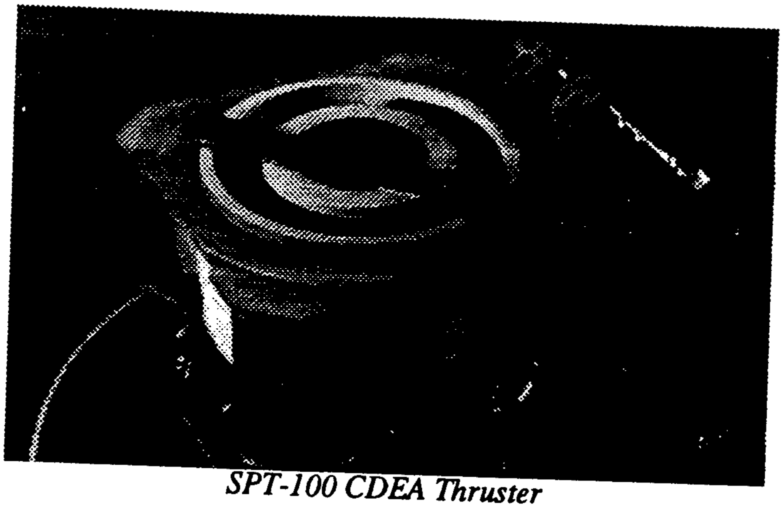
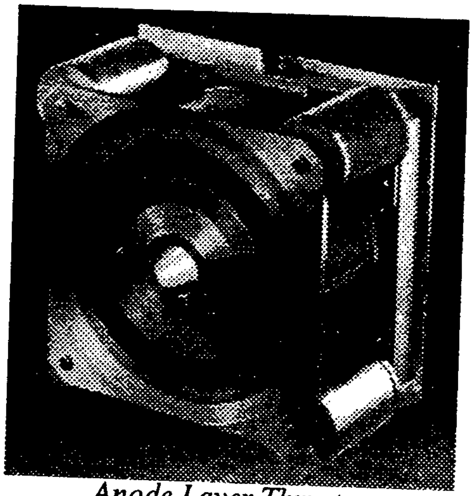


Figure 1.4: Picture of a Typical Arcjet Thruster

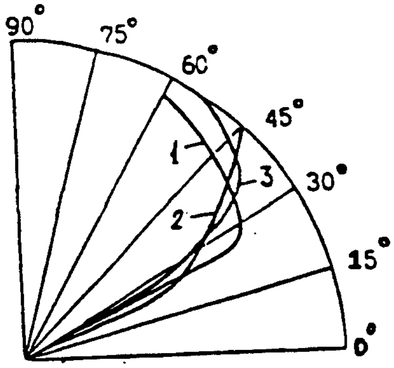


SPT-100 CDEA Thruster



Anode Layer Thruster

Figure 1.5: Pictures of Typical Closed Drift Thrusters



1 - ion current density obtained by experiment
2 - ion current density of "high speed" ions
3 - Full ion current density, obtained numerically

Figure 1.6: Comparison of Results from Bishaev[1993] and Experiment
(As taken from Bishaev [1993])

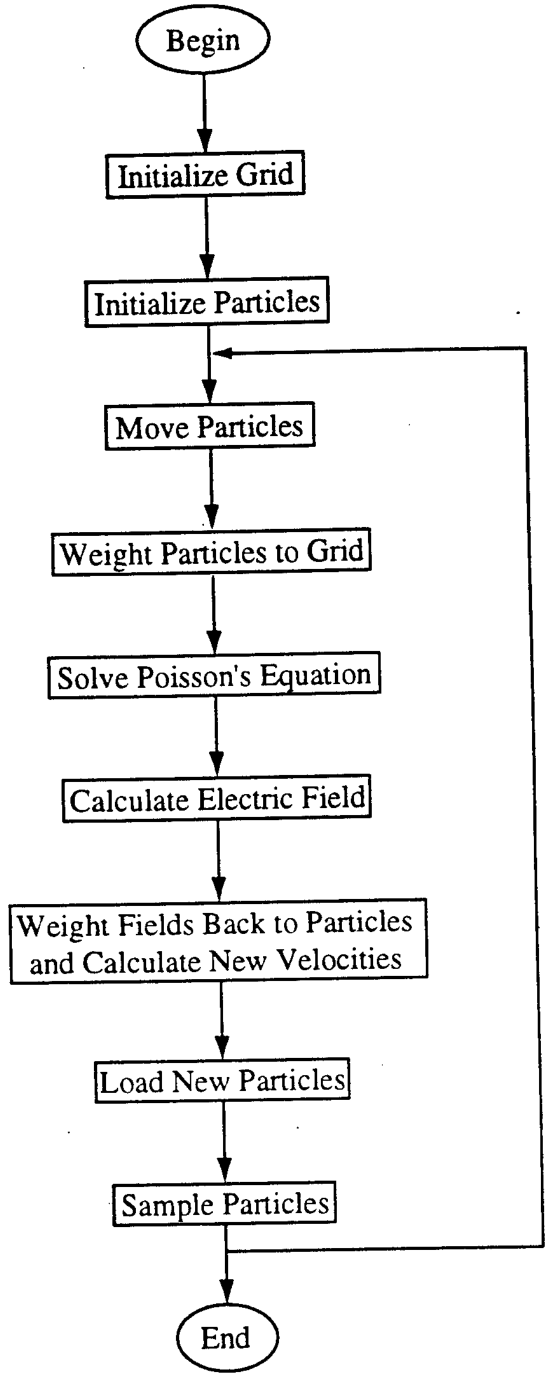


Figure 1.7: Flowchart Diagram of PIC Method

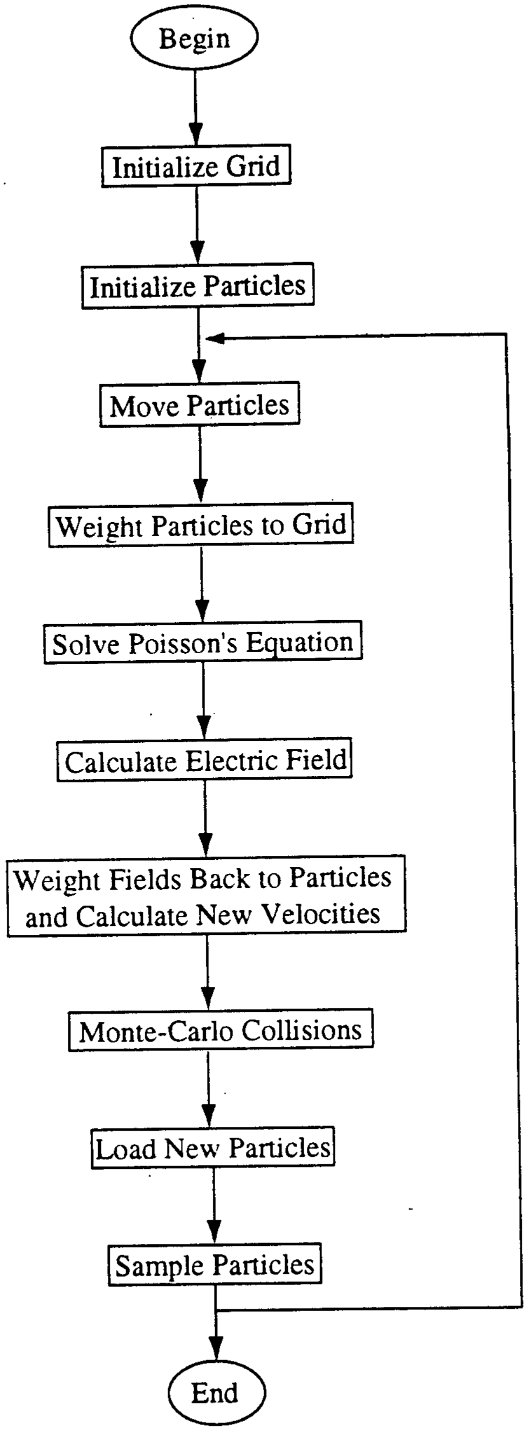


Figure 1.8: Flowchart Diagram of PIC-MCC Method

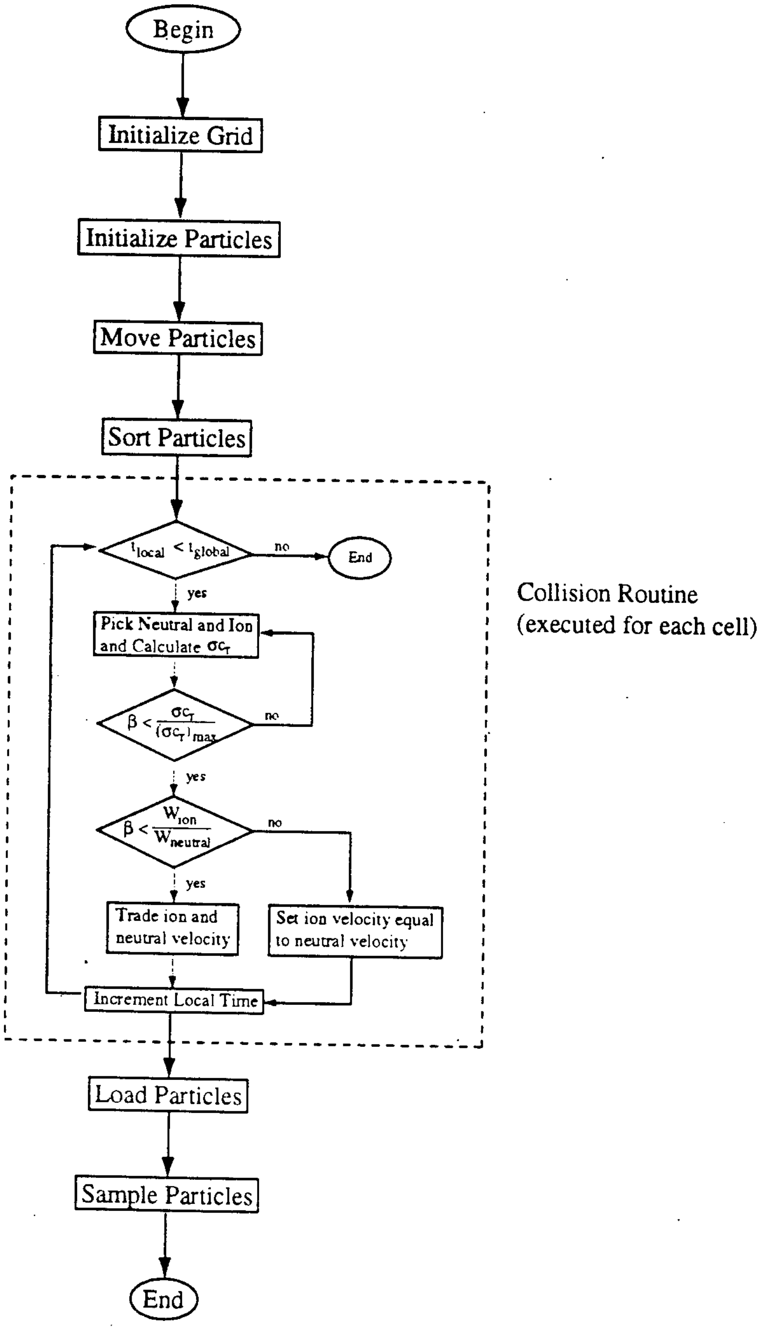

Begin
Initialize Grid
Initialize Particles
Move Particles
Sort Particles
$t_{local} < t_{global}$
no
End
yes
Pick Neutral and Ion and Calculate σc_T
$\beta < \frac{\sigma c_T}{(\sigma c_T)_{max}}$
no
yes
$\beta < \frac{W_{ion}}{W_{neutral}}$
no
yes
Trade ion and neutral velocity
Set ion velocity equal to neutral velocity
Increment Local Time
Collision Routine (executed for each cell)
Load Particles
Sample Particles
End

Figure 1.9: Flowchart Diagram of DSMC Method

Chapter 2: Basic Theory and Analysis

The plume of a Hall Thruster is composed of several different species of charged and neutral particles which interact with themselves and with each other throughout the plume region. The first step in constructing a computational model is to describe the thruster's basic operating characteristics and determine what simplifying assumptions can be made about the plasma. This in turn requires an understanding of the basic characteristics of the plasma, including its Debye length, Gyro radius and Knudsen number. This chapter describes the basic operating characteristics of a Hall Thruster and uses a simple empirical model to describe and estimate the value of fundamental parameters in the plume region. The model used is a point source model (similar to Equation 1.2) that is intended to produce order of magnitude estimates of parameters in the plume region. The model does not accurately duplicate the detailed structure of the plume and should not be used for design purposes. Based on the calculated parameters, a series of simplifying assumptions can be made about the plume region. These assumptions not only clarify the physics of the plasma, but also form the basis for the computational model described in Chapters 3 and 5. Section 2.1 describes in qualitative terms the basic operating characteristics of a Hall Thruster and Section 2.2 presents a simple plume model which is used to estimate various plume parameters. Section 2.3 summarizes the assumptions which can be made based on the calculations presented in Section 2.2. and briefly considers their implications for the computational model. The axisymmetric and three dimensional computational models are described in detail in Chapters 3 and 5 respectively.

2.1 Hall Thrusters: Basic Principles of Operation

The primary difference between a Hall Thruster and a gridded ion thruster is that the ionization and acceleration of propellant take place in a single acceleration channel rather than in separate ionization and acceleration zones. Figure 2.1 is a schematic diagram of a Hall Thruster. The thruster anode region is axisymmetric about the centerline, but the cathode is a point source located to one side of the thruster. The acceleration channel is an annular region surrounded by an insulating dielectric material. Magnetic coils are used to set up a radial magnetic field in the acceleration channel. The strength of this field varies with axial location, but the exact configuration of the field is considered proprietary information. An electric field is established between an annular anode at the closed end of the channel and a hollow cathode placed just to the side and front of the open end of the channel. The potential difference between that anode and the cathode is the "discharge voltage" and is typically 300 V, though SPT thrusters have been operated at higher and lower voltages in the laboratory. The cathode is typically a single hollow cathode, though in principle a different electron source could be used instead. The cathodes used on the SPT-100 are of Russian design and are described in *Arkhipov et al.* [1991]. Electrons emitted by the cathode are attracted towards the anode and enter the acceleration channel. In the channel, the electrons are trapped by the radial magnetic field and begin to orbit around the field lines. At the same time, the interaction of the electric and magnetic fields cause the electrons to drift in the azimuthal direction, creating the so-called "Hall" current (as defined in *Mitchner et al.* [pg. 173]). This current is the source of the term "Hall Thruster." The strength of the magnetic field is such that electrons entering the channel are considered "magnetized" because they gyrate around the field lines while ions entering the channel are considered "unmagnetized" because they follow relatively straight trajectories. The working fluid, typically Xenon gas, flows into the acceleration channel and is impact ionized by the electrons trapped by the magnetic field. Both single and double ions are created by impact ionization. The collisions enable electrons to move towards the anode through a diffusion process. The ionization process is relatively efficient, and 95% of the working fluid is typically ionized in an SPT-100 thruster. The resulting ions are then electrostatically accelerated out the end of the acceleration channel to produce thrust. Because ions are much heavier than electrons, their trajectories are relatively straight and are largely unaffected by the presence of the magnetic field. Nevertheless, a substantial number of ions end up colliding with the walls of the anode. This results in wall losses that are thought to have a significant impact on Hall Thruster efficiency.

Because the ionization process is less than 100% efficient, some neutrals also escape the acceleration channel. The neutrals move at thermal velocities which are much lower than the beam ion velocity. Ions leaving the acceleration channel reach velocities in excess of 17000 m/s while neutrals typically travel at speeds less than 500 m/s. Unlike an ion thruster, however, not all of the ions in the beam of a Hall Thruster travel at the same velocity. Ions created in different parts of the acceleration channel fall through different electrical potentials and end up with different final velocities. Ions created near the end of the channel see relatively little voltage drop and move at low velocities while ions created near the anode see the entire 300 V drop and end up moving at high velocities. As a result, RPA measurements of the beam ions show a much wider range of ion energies than is seen in ion thrusters (see Figure 2.2). An additional source of gas is the hollow cathode, which also uses propellant to create a contained plasma. The hollow cathode acts as an electron source and neutralizes the ion beam released by the thruster. It releases relatively few ions into the plume. However, the hollow cathode is a substantial source of neutral flow and must be considered in any comprehensive plume model.

In terms of composition, the plume of a Hall Thruster bears some resemblance to the plume of an ion thruster. In both cases, a group of slow moving neutrals interacts with a group of fast moving ions through charge exchange collisions to create charge exchange plasma. However, some critical differences exist between the plumes from these two devices. The mass flow rate through an ion thruster is about one tenth of the flow rate through a Hall Thruster and velocity of the ions is about 2.3 times faster in the plume region [*Samanta Roy* 1995, pg. 45]. The peak electron number density in an ion thruster plume is therefore much lower than that in a Hall Thruster, about 1 x 10^{16} m^{-3}. The neutral density in an ion thruster is about the same as that in a Hall Thruster, so the neutral density is over 100 times higher than the ion density [*Samanta Roy* 1995, pg. 25-28]. Because the neutral density is much greater than the plasma density, hybrid PIC-MCC methods can be used to simulate the plumes of ion thruster. The peak electron number density in a Hall Thruster plume is about 2 x 10^{17} m^{-3}, or ten times higher than the peak plasma density in an ion thruster. The higher electron density lowers the Debye length by a factor of 3 when compared to ion thrusters, so thirty times more cells are required to simulate a Hall Thruster in three dimensions than are required to simulated an ion thruster plume of the same size (though not the same power or thrust). Given that previous ion thruster simulations have required massively parallel supercomputers [*Samanta Roy* 1995], the larger number of cells required poses a practical barrier to the use of conventional PIC techniques. An additional problem is that the neutral density in a Hall Thruster can be of the same order as the plasma density. Because the PIC-MCC method assumes that the plasma density is much higher than the neutral density, it does not apply in this situation.

It should be noted that Figure 2.1 presents a simplified picture of Hall Thruster operation. Experimental and analytical work has shown that a considerable number of instabilities occur in the acceleration channel, creating significant time dependent variations in potential and charge density [*Dickens et al.* 1995]. Analytic and experimental work also suggests that the magnetic field gradient at the end of the acceleration channel plays a role in maintaining the stability of the discharge [*Bishaev et al.* 1978]. Because this work is more concerned with time average effects in the plume than with the physics of the discharge itself, we choose to ignore these time dependent effects. Further reading on time dependent phenomena is available in *Dickens et al.* [1995] and *Fife* [1995].

Anode layer thrusters like the TAL thruster operate on the same principles as the Hall Thruster. A cathode and acceleration channel are present in anode layer devices, but the acceleration channel is lined with metal rather than dielectric material and is much shorter than the channel of an SPT-100. Because the acceleration channel is quite short, most of the ion acceleration is thought to occur in a thin region in front of the acceleration channel rather than inside the channel itself. This region is referred to as an "anode layer" and is the source of the term "anode layer thruster." The exact manner in which a TAL thruster operates is presently unknown. In particular, it is unclear what, if any, role the conductive surfaces of the anode play in the ionization and acceleration process.

2.2 Fundamental Parameters

The best studied of the closed drift thrusters is the SPT-100, which is a type of Hall Thruster. The SPT-100 has been studied extensively in Russia and in the West and its basic characteristics are well documented. The SPT-100's operating parameters are summarized in Table 2.1 and a picture of the SPT-100 is shown in Figure 1.5. The values shown in Table 2.1 are nominal, and conditions may vary from thruster to thruster and experiment to experiment. Experiments have been conducted on SPT's operating at different discharge voltages, mass flow rates, and even when using different propellants [*Marrese* 1995].

Specific Impulse[5]	1610 sec.
Specific Impulse, Anode only[5]	1735 sec.
Thrust[5]	84.9 mN
Discharge Voltage[5]	300 V
Discharge Current[5]	4.5 A
Power[5]	1350 Watts
Efficiency[5]	49.7%
Efficiency, Anode only	53.5%
Inner Insulator Diameter[1]	56 mm
Outer Insulator Diameter[1]	100 mm
Propellant	Xenon
Propellant Flow Rate[2]	5.0-5.2 mg./sec.
Fraction of Propellant Directed to Cathode[2]	~10%
Electron Temperature in Plume[3] (T_e)	2-4 eV
Axial Ion Velocity[4] (v_z)	~17000 m/s
Fraction of Propellant Ionized in Discharge Chamber [2] (η_i)	> 95%

Table 2.1: SPT-100 Basic Characteristics

"Anode only" indicates that the cathode flow is excluded.

[1] [Manzella 1994], [2] [Absalamov 1993], [3] [Myers and Manzella 1993], [4] [Manzella 1993], [5] [Manzella and Sankovic 1995]

Although the values shown in Table 2.1 are specific to the SPT-100 thruster, they are "typical" values for generic Hall and anode layer thrusters.

The fundamental characteristics of the plume region can be estimated based on Table 2.1 and some simple assumptions. From flux conservation, the ion and neutral density at the end of the acceleration channel is given by

$$n_i = \eta_i \dot{m} / v_z A$$

$$n_o = (1 - \eta_i)\dot{m} / a_o A \qquad (2.2\text{-}1)$$

Where $\dot{m}$ is the mass flow through the device, A is the exit area and a_0 is the speed of sound. The neutrals can be assumed to be choked at the exit of the acceleration channel, so their drift velocity is given by the local speed of sound

$$a_o = \sqrt{\frac{\gamma k T}{m}}$$

For a monatomic gas like Xenon, $\gamma = 5/3$. The temperature of the neutrals in the acceleration channel has not been measured, but 0.1 eV is a reasonable guess and gives a sonic speed of 350 m/s. From Table 2.1, the fraction of the total flow which is directed to the anode is 90%, the ionization fraction at the end of the channel is approximately 95%, the inner acceleration channel radius is 23 mm and the outer channel radius is 50 mm. Using these values in equation (2.2-1) gives $n_i = 2.22 \times 10^{17}$ m^{-3} and $n_o = 5.69 \times 10^{17}$ m^{-3} at the anode exit. The calculated plasma density is consistent with near-field measurements of the electron number density 1 cm downstream of the exit plane [*Kim et al.* 1978]. The neutral density has not been measured in this region.

Equation (2.2-1) can also be used to estimate the plasma and neutral density at cathode exit. A total of 10% of the propellant flowing to the SPT-100 is directed to two hollow cathodes placed just in front and to the side of the thruster. Only one cathode is needed to operate the thruster, but two are used to provide redundancy in case of cathode failure. In the past, both cathodes continuously received propellant even though only one was operational at any point in time. This procedure caused erosion to occur on the inactive cathode and may no longer be used. The ionization fraction in a hollow cathode is relatively low (1%-10%) and can be approximated as zero for purposes of this analysis. The cathode orifice itself is very small, typically 0.5 mm [*Williams et al.* 1990]. Using these values in equation (2.2-1) gives a neutral density $n_o = 4.34 \times 10^{21}$ m^{-3} at the cathode exit.

Because the plume is expanding spherically, it is reasonable to assume that the density will fall as $1/r^2$. This allows one to estimate the density in the plume based on a model like that given in equation (1.1-1). For purposes of this analysis, it is useful to set $f(\theta) = 1$. Although the plume actually has higher densities along the centerline and lower densities at the edges, $f(\theta)$=constant is sufficient for estimating the plume's characteristics. If the acceleration channel were a point source, A would be equal to the radius of the orifice. Since the channel is actually annular, A is set equal to the width of the acceleration channel. This provides a rough estimate of the density in the plume region. A similar model can used to estimate the density contribution from the cathode. In this case, $f(\theta) = 1$ and A is set equal to the radius of the cathode orifice. The resulting plasma and neutral densities are shown in Figure 2.3. The contributions from the anode and cathode are shown separately and can be superimposed to obtain the local neutral density.

The plasma densities shown in Figure 2.3 are generally of the same order as experimental measurements of plasma density [*Myers and Manzella* 1993]. As one would expect, Figure 2.3 underpredicts the plasma density along the centerline and therefore gives only a rough approximation of conditions in the plume. No direct experimental measurements have been made of the neutral density in the plume region. A retarding pressure sensor has been used to measure the neutral flux 0.5 m from the anode exit. The measured flux varied from 1×10^{20} to

1 x 10^{22} m^{-2} sec^{-1} [*King* 1996], assuming that the neutrals are moving at a thermal velocity of 350 m/s gives number densities ranging from 3 x 10^{17} to 3 x 10^{19} m^{-3}. This value is higher than the densities shown in Figure 2.3, and the reason for the disagreement is unclear. One possibility is that the neutrals in the plume are actually fast neutrals produced by CEX. The experimental technique used to measure the neutral flux is not well established, so further work is needed to clarify these measurements and accurately determine the neutral density in the plume region.

A variety of fundamental parameters can be calculated from the information shown in Figure 2.3. Table 2.2 gives a list of fundamental parameters calculated based on Figure 2.3. For simplicity, the flow from the cathode is not included in these calculations. Table 2.2 gives a rough approximation of conditions in the plume. These parameters and the methods used to calculate them are discussed in sections 2.2.1 to 2.2.3 below.

	Radial Position		
	0.2 m	0.5 m	1.0 m
n_i,n_e	2.69 x 10^{15} m^{-3}	4,31 x 10^{14} m^{-3}	1.08 x 10^{14} m^{-3}
n_o	6.88 x 10^{15} m^{-3}	1.10 x 10^{15} m^{-3}	2.75 x 10^{14} m^{-3}
B	~1 G	~ 0.1 G	~0.01 G
λ_d	0.02 cm	0.05 cm	0.10 cm
$\rho_{electron}$	5.4 cm	54 cm	540 cm
$\rho_{beam\ ion}$	232 m	2320 m	>20000 m
$\rho_{cex\ ion}$	4.8 m	48 m	480 m
λ_{en}	908 m	5680 m	22700 m
λ_{ee}	26 m	160 m	660 m
λ_{ei}	35 m	220 m	880 m
λ_{in}	2980 m	18700 m	> 70000 m
λ_{ii}	109 m	685 m	2740 m
λ_{CEX}	242 m	1510 m	6060 m
λ_{nn}	395 m	2470 m	9890 m

Table 2.2: Estimates of Fundamental Parameters in the Plume Region

2.2.1 Debye Length

The characteristic length over which charges are neutralized in a plasma is given by the Debye length

$$\lambda_d = \sqrt{\varepsilon_o k T_e / n_e e^2}$$

Table 2.2 shows the Debye length at various distances from the thruster based on an electron temperature of 2 eV. Throughout the plume the Debye length is small with respect to features of interest. Even at distances up to 4 m from the thruster exit, the Debye length is only half a centimeter, which is very small compared to a meter-scale spacecraft. Since there is no experimental evidence for double layers or other free standing sheaths, the plume can be considered quasi-neutral everywhere except in a thin region near solid surfaces. Because the Debye length is so small, it is impractical to model the plume using conventional PIC methods. However, since the Debye length is small everywhere in the plume, quasi-neutrality can be used to simplify our PIC model and avoid conventional length scale limitations. This point will be discussed further in Chapter 3.

2.2.2 Magnetic Parameters

A radial magnetic field is used to capture electrons in the acceleration channel and encourage impact ionization of the propellant. The extent to which this magnetic field "leaks" into the plume region has been measured experimentally, and is shown in Figure 2.5. Superimposed on the data in Figure 2.5 is a dipole fit to the magnetic field data corresponding to a $1/r^3$ power law. The dip in the data is caused by a reversal in the sign of B which can not be directly shown on a log plot. A $1/r^3$ fit corresponds to +/- 20% of the data and shows that the B field falls very rapidly in the plume region. It should be noted that the geomagnetic field in GEO is of order 0.003 G, so the field from the thruster own field dominates throughout most of the plume region. This section describes methods used to calculate parameters relating to the magnetic field and its influence on the plasma.

Magnetic Pressure Ratio

The importance of the magnetic field in a plasma is measured by two critical parameters. The first is the ratio of the kinetic energy in the plasma to the magnetic energy in the plasma, which is given by

$$\beta_p = \frac{\sum nkT}{B^2/2\mu_o}$$

The numerator is summed over the all of the charged species in the plasma; the neutrals have no effect on the strength of the magnetic field. A plasma cloud is diamagnetic and tends to neutralize magnetic fields. When beta is large, the plasma pressure dominates over the magnetic pressure and the plasma effectively shields out external magnetic fields with its own self generated fields. When beta is small, external fields are unmodified by the presence of the plasma. At the anode outlet, beta is approximately given by

$$\beta = \frac{(2.2\times10^{17}\,\mathrm{m}^{-3})(1.38\times10^{-23}\,\mathrm{J/K})(2\ \mathrm{eV})(11600\ \mathrm{K/ev})}{(0.013\ \mathrm{T})^2/2(1.26\times10^{-6}\ \mathrm{H/m})} = 0.070$$

The resulting value is small as one would expect, or the thruster wouldn't operate. As one moves away from the thruster, the magnetic field falls as $1/r^3$ while the density falls as $1/r^2$. As a result, the local value of β_p rapidly increases, reaching unity at a distance of about 0.2 meters from the channel exit. As shall be shown in the next section, however, by the time β_p reaches unity, the local magnetic field has become very weak and has only a marginal influence on the plume.

Electron Gyro Radius

The second parameter which measures the importance of the magnetic field in a plasma is the gyro radius, which is given by

$$\rho_{gi/e} = \mathrm{m}_{i/e}\mathrm{v}_{\perp} / \mathrm{eB}$$

$\mathrm{v}_{\perp}$ is the component of the particle's velocity perpendicular to the magnetic field and varies with the orientation of the local magnetic field. Because ions and electrons have different masses and temperatures, the electron and ion gyro radii are considered separately in this analysis. Experimental measurements indicate that the electron temperature is nearly constant throughout the plume region (see Figure 2.4). Electrons in the plume can therefore be treated as an isothermal gas. We further assume that the electrons are Maxwellian, so their mean velocity is given by

$$\bar{c} = \sqrt{\frac{8kT}{\pi m}}$$

Maxwellian electrons with a temperature of 2 eV have a mean speed is 9.52 x 10^5 m/s. The gyro radius has been calculated using this value for $\mathrm{v}_{\perp}$ and is shown in Table 2.2. Although the electron gyro radius is small inside the discharge chamber, it becomes quite large 20 cm from the thruster and much larger than features of interest when 1 m away from the thruster. In fact, the electrons are effectively unmagnetized at distances > 25 cm from the channel exit, which suggests that it may be possible to ignore the magnetic field in some regions of the plume. It should also be noted that some experimental work suggests that the structure of the plume is not influenced by small changes in the magnetic field [*Manzella et al.* 1995]. Based on the theoretical calculations and these experimental observations, we choose to treat the electrons as unmagnetized throughout the plume region. This assumption is clearly valid when more than 25 cm from the thruster, but may not be valid in the near field regions near the channel exit. The implications and limitations of this assumption are discussed further in Chapter 3.

Ion Gyro Radius

RPA measurements of ions in the plumes of Hall Thrusters have shown that they are highly non-Maxwellian [*Absalamov* 1992]. However, previous work with ion thrusters suggests that the ions in the plume can be divided roughly into two groups: beam ions, which move at velocities of around 17000 m/s, and CEX ions which move at thermal speeds. Because these ions travel at different speeds, they are characterized by two different gyro radii. The beam ion gyro radius (given as $\rho_{\text{beam ion}}$) has been calculated based on a beam ion velocity of 17000 m/s. The results, shown in Table 2.2, indicate that the trajectories of these ions are clearly unaffected by the presence of the magnetic field. This result is to be expected since the magnetic field is designed to trap electrons in the acceleration channel without trapping the ions created in the channel itself. CEX ions were treated as a Maxwellian gas with a temperature of 0.1 eV, which is equal to the expected neutral temperature This value is probably low since the CEX ions are typically accelerated to an energy of several volts after they are created, so the calculated ion gyro radius is conservative and probably smaller than the real CEX ion gyro radius. The CEX ion gyro radius is also shown in Table 2.2, and the results again show that the CEX gyro radius is much larger than features of interest in the plume. It is therefore safe to conclude that ions in the plume are basically unmagnetized throughout the plume region.

2.2.3 Mean Free Paths

The importance of collisions is measured by the mean free path, which is given by the well known relationship

$$\lambda = 1/(n\sigma) \qquad (2.2\text{-}2)$$

In an SPT plume, there are three distinct species of particle: ions, neutrals, and electrons. Each species undergoes a variety of interactions with itself and other species, each of which is characterized by a collision cross section and a mean free path. The possible interactions include elastic collisions, CEX collisions, ionization, excitation, recombination, and many others [*Mitchner et al.* pg. 15]. This section describes the methods used to calculate collision cross sections and mean free paths of several significant processes which play a role in the development of the plume from a Hall Thruster.

Electron Mean Free Paths

The total electron-Xenon neutral collision cross section has been determined experimentally and is shown in Figure 2.6 [*Chapman* pg. 45]. For electrons with an average velocity of 2 eV, the total cross section is approximately 1.6 x 10^{-19} m^2. The resulting electron-neutral mean free path has been calculated based on the plume model presented in Section 2.2 and is given in Table 2.2.

These values are large with respect to the plume, indicating that as long as the electrons are unmagnetized, electron-neutral collisions can be effectively neglected throughout the plume region.

The electron-electron and electron-ion collision cross sections are based on Coulomb interactions and can be calculated from theory. Coulomb collision cross sections can be calculated from the Rutherford cross section, as described in *Mitchner and Kruger* [1973]. The Rutherford differential scattering cross section is given by

$$I(\chi, c_r) = \frac{(b_o/2)^2}{\sin^4(\chi/2)} \qquad (2.2\text{-}3)$$

Where *I* is the differential scattering cross section and b_O is the impact parameter for 90 degree scattering as defined in Figure 2.7. The impact parameter depends on the interaction potential, which is this case is the electrostatic potential

$$\Phi(r) = \pm\frac{A}{r} \qquad A = \frac{Z_1 Z_2 e^2}{4\pi\varepsilon_o} \qquad (2.2\text{-}4)$$

Where r is the distance between the particles. Given (2.2-4), the impact parameter for 90 degree scattering is given by

$$b_o = \frac{A}{\mu g^2}$$

Integrating (2.2-3) then gives the momentum transfer cross section.

$$\sigma^{(1)}(c_r) = 4\pi b_o^2 \ln\left[1 + \left(\frac{\lambda_d}{b_o}\right)^2\right]^{½} \qquad (2.2\text{-}5)$$

Integrating (2.2-5) over energy gives the mean cross section. Banks gives the result of integrating (2.2-5) for two species at different temperatures.

$$\bar{\sigma} = \frac{\pi}{2}\left(\frac{Z_1 Z_2 e^2}{\mu}\right)^2 \frac{\ln\Lambda}{\left(\frac{kT_1}{m_1} + \frac{kT_2}{m_2}\right)^2}$$

For ions and electrons with $T_e/m_e >> T_i/m_i$ this reduces to (in cgs units)

$$\bar{\sigma}_{ie} = \frac{\pi}{2}\frac{(Z_i e^2)^2 \ln\Lambda}{(kT_e)^2} \qquad (2.2\text{-}6)$$

ln Λ is the Spitzer logarithm which is a tabulated quantity that varies slowly with density. 13 is a reasonable value for the range of densities and temperatures in the plume of a Hall Thruster [*Mitchner et al.* pg. 59]. The electron temperature in the plume is 2 eV, so the cross section works out to 1.06 x 10^{-17} m^2. Table 2.2 gives the resulting electron-ion mean free path. Again, these values are extremely large when compared to features of interest, so electron-ion interactions do not need to be considered in the plume region.

The mean momentum transfer Coulomb collision cross section for singly ionized species is given by *Mitchner et al.* [pg. 58]

$$\overline{\sigma}_{ee} = 6\pi b_0^2(\ln\Lambda) \cong \left(5.85\times10^{-10}\right)\ln\Lambda / T_e^{\,2}\ m^2 \tag{2.2-7}$$

This expression is a modified form of equation (2.2-6) and can be used to calculate the electron-electron mean free path. For an electron temperature of 2 eV, the collision cross section works out to σ = 1.413 x 10^{-17} m^2. The resulting electron-electron mean free paths are listed in Table 2.2. These values are also much larger than features of interest in the plume.

In summary, the electron-neutral, electron-ion, and electron-electron mean free paths are all long with respect to features of interest in the plume region. Therefore, in the absence of a magnetic field, electrons are effectively collisionless in the plume region.

Ion Mean Free Paths

The Xe-Xe^+ cross section for elastic collisions has not been measured experimentally, but can be estimated from theory. *Samanta Roy* [1995] used the following approximation from *Banks* [1966] (in cgs units)

$$\sigma_{in}(c_r) = 2.21\pi\left(\frac{\psi e^2}{m_{12}c_r^{\,2}}\right)^{1/2} \tag{2.2-8}$$

Where c_r is the approach velocity, ψ is the polarizability of the neutral atom, and m_{12} is the reduced mass of the two particles. The polarizability of Xenon ψ is 4.044 x 10^{-24} cm^3. The beam ions are traveling at 17000 m/s. Since the neutrals are generally moving at velocities less than 500 m/s, the ion-neutral approach velocity is essentially equal to the beam ion velocity and the ion-neutral collision cross section works out to 4.87 x 10^{-20} m^2. Once the collision cross section is known, equation (2.2-8) can be used to calculate the ion-neutral mean free path. The results are shown in Table 2.2. As before, the ion-neutral mean free path is large with respect to features of interest. It should be noted that collisions between ions and neutrals can also result in the creation of new ions through impact ionization. The cross section for impact ionization is also of order 1 x 10^{-20} m^2, so the mean free path of this process will also be long with respect to features of interest [*Chapman*, pg. 42].

Like electron-electron collisions, ion-ion interactions are governed by Coulombic forces. The ion-ion collision cross section is therefore given by the same expression used to electron-electrons collisions, equation (2.2-8). The ion temperature is subject to some uncertainty. The axial ion temperature measured by Manzella [1994] is 3.4 eV. The results shown in Table 2.2 were calculated on the basis of this temperature. Again, the ion-ion mean free path is large with respect to features of interest. Ion-ion collisions can therefore be ignored in the plume region.

One additional process of interest is charge exchange collisions. The cross section for CEX between Xenon neutrals and singly ionized Xenon ions has been measured experimentally and is shown in Figure 2.8. The relative velocity between the neutrals and ions is approximately 17000 m/s, which results in a cross section of 6.0 x 10^{-19} m^2. Table 2.2 shows the resulting mean free paths. Although the resulting mean free paths are large, the backflow which results from CEX collisions is non-negligible and must be considered in any analysis of the plume region. The reason for this bears some discussion. While most elastic collisions serve mainly to thermalize the plasma, each CEX collision creates a slow ion which can be accelerated sideways and backwards by the potential structure of the plume region. When even a small percentage of the beam undergoes CEX, a significant backflow results. Ion-neutral collisions also have the potential to create slow ions, but momentum is exchanged less efficiently in these collisions and the ion-neutral mean free path is an order of magnitude larger than the CEX mean free path. As a result, CEX collisions dominate the creation of slow ions and must be included in our computational model.

Neutral Mean Free Paths

The Xe-Xe elastic collision cross section has been measured experimentally. The *Handbook of Chemistry and Physics* [*Lide* pg. 6-240] gives the molecular diameter of room temperature Xenon as 3.42 x 10^{-8} cm, which corresponds to a collision cross section of 3.674 x 10^{-19} m^2. This value has been used to calculate the neutral-neutral mean free paths (designated as λ_{nn}) in Table 2.2. The resulting mean free paths are large with respect to the plume, indicating that the neutral component of the gas is largely collisionless as it leave the thruster. It is worth noting that λ_{nn} is of the same order as λ_{CEX}, but since Xe-Xe collisions serve mainly to thermalize the neutrals the few collisions that do occur in the plume have little effect on the final distribution.

2.3 Summary

A simple plume model has been used to calculate approximate values for various fundamental plasma parameters in the plume of an SPT-100 thruster. The results are based on laboratory measurements of the magnetic field in the plume, a simple $1/r^2$ plume expansion model, and a series of different models and measurements of collisions cross sections for ions, electrons, and neutrals in the plume region. The plume was assumed to be composed of Xenon ions, Xenon neutrals, and electrons and was shown to be quasi-neutral throughout the plume region. Based on laboratory measurements, the electrons were assumed to be isothermal. The results of these calculations are summarized in Table 2.2. Based on the work presented in this chapter, the following statements can be made about the plume from a typical Hall Thruster.

- The Debye length, is small, so the plume is quasi-neutral except in a small region near solid surfaces.
- The ions are unmagnetized throughout the plume.
- The electrons are unmagnetized when $z > 25$ cm.
- The electrons are collisionless.
- The neutrals are collisionless.
- CEX is the dominant ion collision process.

The picture which emerges is that of an unmagnetized, quasi-neutral plasma in which CEX collisions play a significant role but in which the electrons are collisionless. These observations form the basis for a computational model of the plume based on the PIC and DSMC methods. This model, and its underlying assumptions and limitations are discussed in detail in Chapter 3.

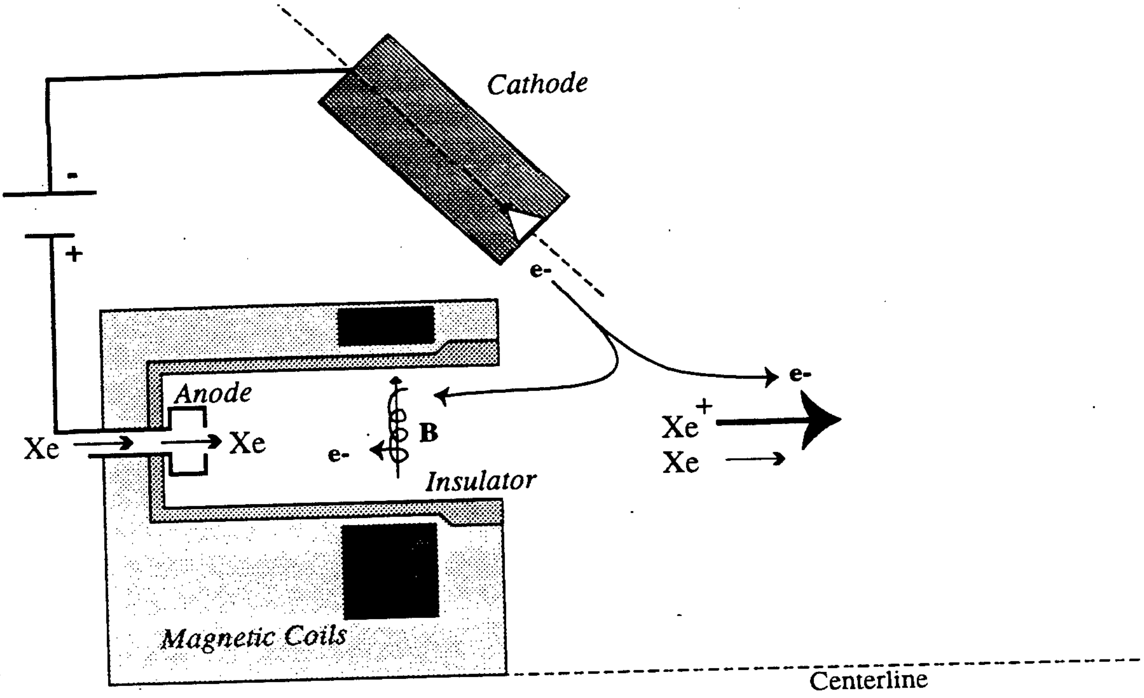


Figure 2.1: Schematic Diagram of a Hall Thruster

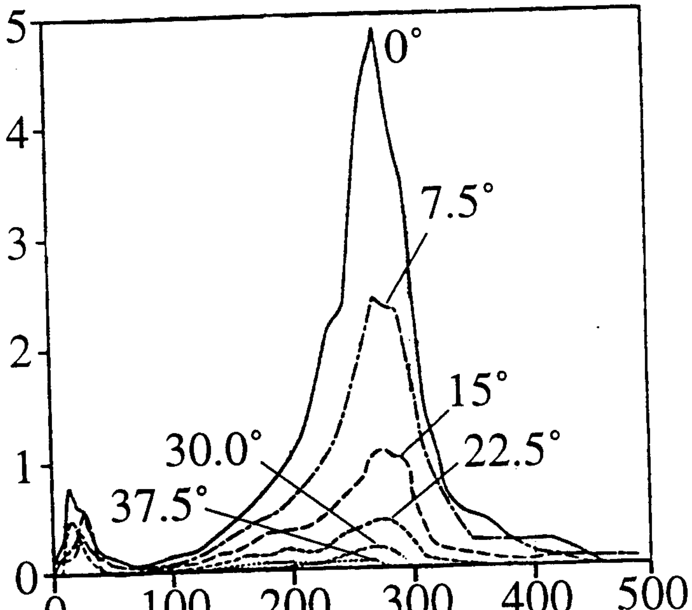
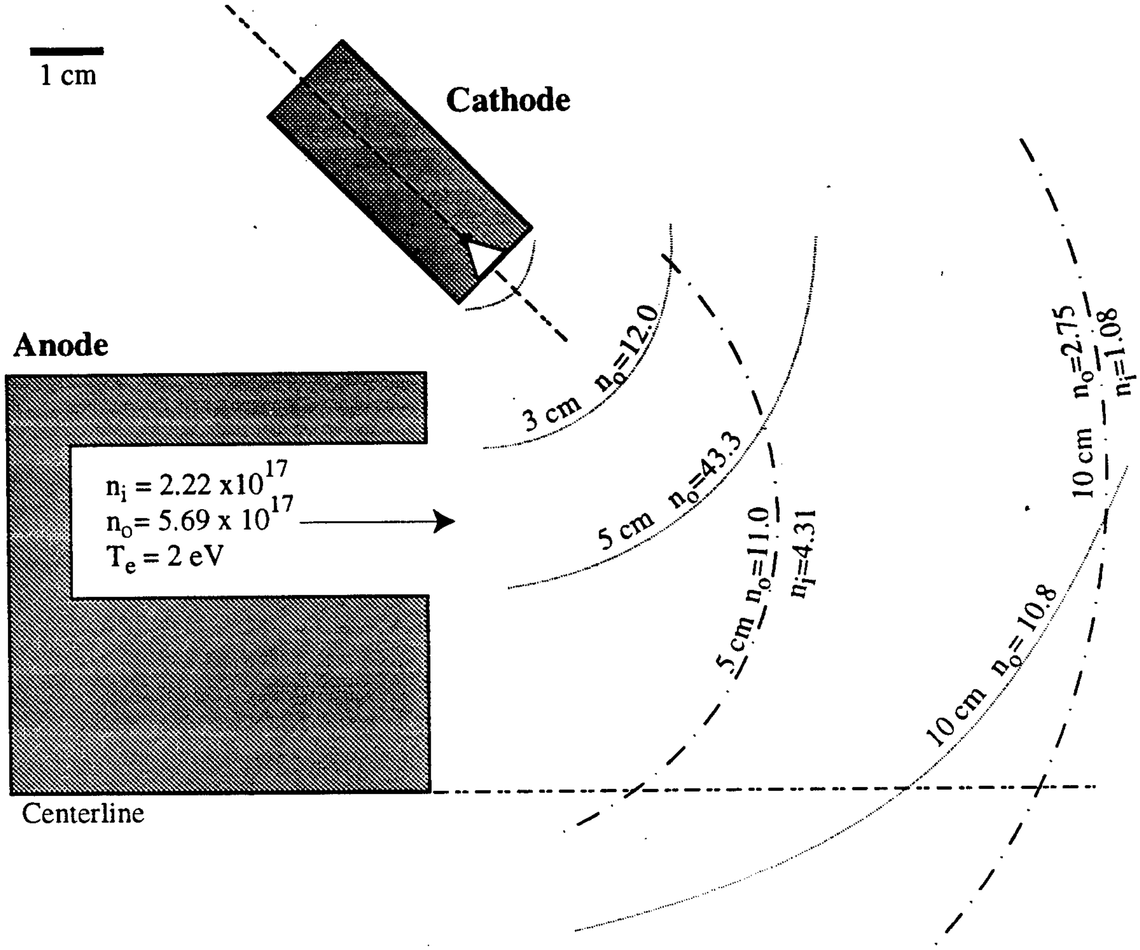


Figure 2.2: RPA Measurements of Beam Ion Energies
[Absalamov 1992]



Drawing is only approximately to scale.
Contours indicate density in units of 10^{16} m^{-3}

Figure 2.3: Estimates of Neutral and Ion Densities around a Hall Thruster

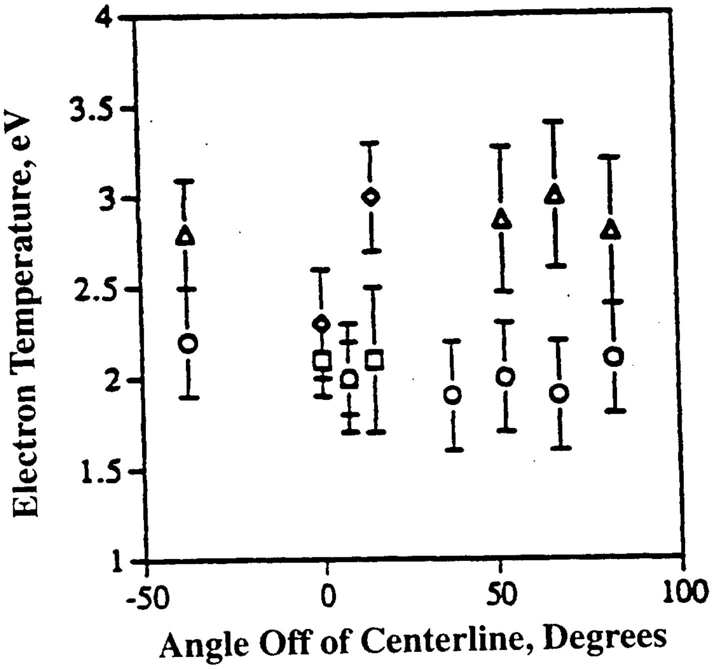


Figure 2.4: Experimental Measurements of the Electron Temperature
Electron Temperature at 2 m and 4 m distance [Myers and Manzella, 1993]
2 m = Circles and Triangles
4m = Squares and Diamonds

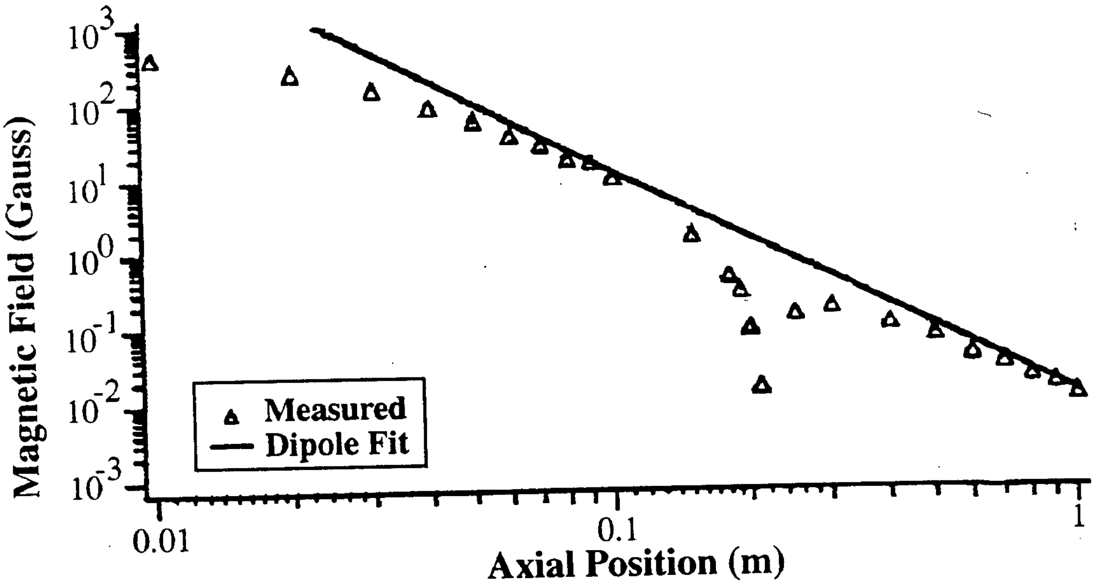
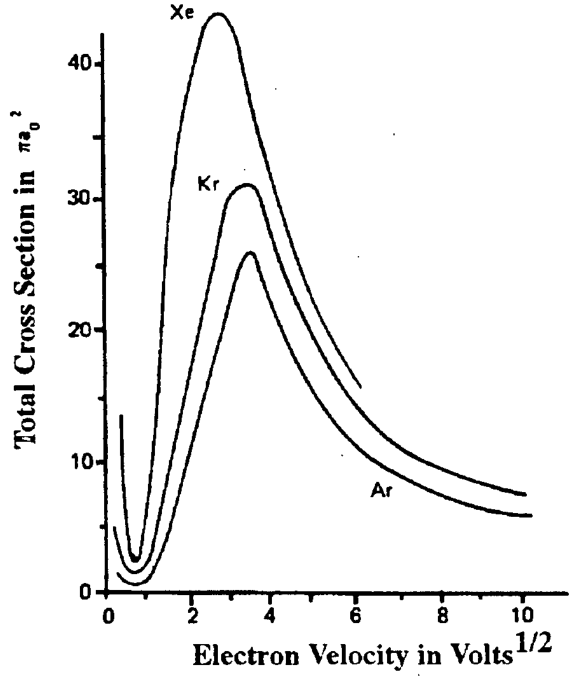


Figure 2.5: Measurements of Magnetic Field Strength vs. Axial Position
[Randolph, SS/L, 1994]



a_0 is the Bohr Radius, so πa_0^2 is 8.82×10^{-21} m^2.

Figure 2.6: Total Electron Collision Cross Section
*[*Chapman *pg. 45]*

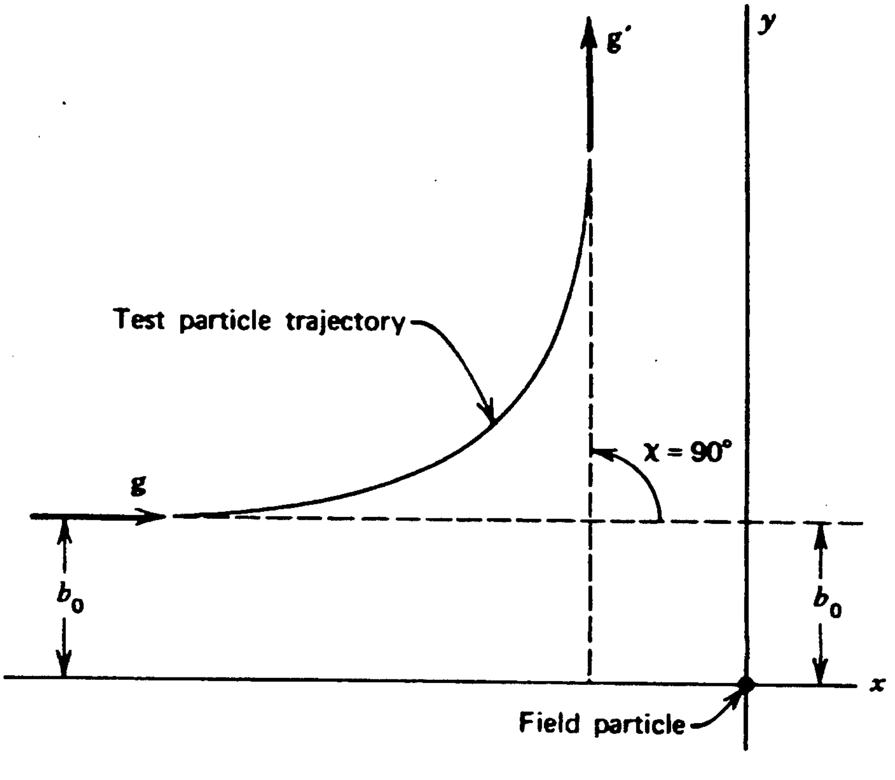


Figure 2.7: Definition of Impact Parameter for 90 Degree Scattering
[Mitchner et al. pg. 55]

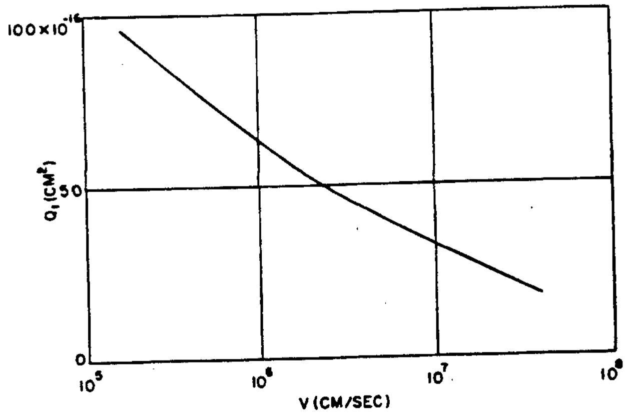


Figure 2.8: Measured Cross Section for Xe-Xe$^+$ CEX
[Brown, pg. 75]

Chapter 3: Computational Plume Model and Methods (Axisymmetric)

In Chapter 2, it was shown that the plume from a Hall Thruster is an unmagnetized, quasi-neutral plasma in which CEX collisions are known to play a significant role. This plume is relatively difficult to simulate because the ion density is high compared to that in an ion thruster. As a result, conventional PIC based methods become too computationally intensive to apply to these devices. In addition, because the ions are highly non-Maxwellian, conventional hydrodynamic models also do not apply in the plume region. Finally, because the neutral and ion densities are of the same order, conventional PIC-MCC methods may not apply in the plume. We use a new computational algorithm to model the plume from a Hall Thruster. This method is a combination of the well known PIC and DSMC methods and uses quasi-neutrality to avoid the length and time scale limitations imposed on conventional PIC models. Combining these two well known methods allows one to track the trajectories of ions and neutrals undergoing collisions in a relatively dense plasma plume. Chapter 3 describes the basic algorithm and gives the details of its implementation in an axisymmetric geometry. A three dimensional PIC-DSMC model has also been developed and is discussed in Chapter 5.

3.1 Introduction

A Hall Thruster plume is a rarefied gas in which both Coulomb forces and collisional effects have a significant impact on the development of the flow. As was discussed in Chapter 1, the Particle in Cell (PIC) method can be used to model long range Coulomb forces, but not collisional effects. The Direct Simulation Monte Carlo (DSMC) method can be used to model collisions in rarefied neutral flows, but can not be used to model the charged particles present in a plasma. These two well known particle simulation methods have been combined to create a hybrid PIC-DSMC simulation. The hybrid simulation has a capability to model moderately collisional non-Maxwellian plasmas interacting with neutral gases. A diagram of the combined PIC-DSMC method is shown in Figure 3.1. The basic method is very similar to the PIC method shown in Figure 1.7. Two important differences are present. First, the potential is determined using the electron momentum equation rather than by solving Poisson's equation. Second, a collision routine is included which uses DSMC methods to modify the trajectories of particles in the plasma. Chapter 3 gives a detailed description of the combined PIC-DSMC algorithm and its implementation in an axisymmetric geometry. The chapter describes the underlying theory and limitations of the algorithm and the practical issues involved in implementing it on a workstation class machine. The chapter is divided into several sections. Sections 3.2 and 3.3 give general background and describe the domain and boundary conditions used in the simulations. Sections 3.4 and 3.5 describe our specific implementation of the PIC and DSMC portions of plume model. Section 3.6 describes the surface interaction model, and section 3.7 discusses computational issues related to memory management on modern (1996) computer workstations.

3.2 Background and Fundamental Assumptions

3.2.1 Normalization of Variables

When conventional SI units are used for numerical analysis, large roundoff errors can occur when very large numbers (2.5 x 10^{17} m^{-3}) and very small numbers (1.6 x 10^{-19} Coulombs) are included in a single calculation. This occurs because machine accuracy is larger than the smallest numbers that can be represented using the IEEE standard floating point numbers (as discussed in *Press et al.* [1992, pp. 28-31]). To avoid roundoff errors, we use normalized (unit-less) variables for most calculations. Input and output files may also use normalized variables, as described in the User's Manual (Appendix A). All of the normalizations in the simulation are based on four basic quantities. These quantities are defined in the program and can be modified, though most users will want to treat them as constants. The four basic quantities and their default values are

- *Charge*: 1.6 x 10^{-19} C (elementary charge)
- *Density*: 1.0 x 10^{12} m^{-3} (arbitrary choice)
- *Mass*: 2.181063 x 10^{-25} kg (weight of a Xe atom)
- *Temperature*: 2 eV (same as plume electron temperature)

Based on these assumed quantities, a variety of different normalizations have been derived. These quantities are defined in the following list. Their default values are also shown (as derived from default values for the reference quantities).

- *Charge/Mass Ratio*:

  -Definition: q_{ref}/m_{ref} -Value: 734,840.5

- *Length*: Debye Length

  -Definition: $\lambda_d = \sqrt{\frac{kT_{ref}\varepsilon_o}{n_{ref}e^2}}$ -Value: 0.010513 m

- *Time*: Inverse of Ion Plasma Frequency

  -Definition: $\omega_{pi} = \sqrt{\frac{n_{ref}e^2}{m_{ref}\varepsilon_o}}$ -Value: 115313.21 sec^{-1}

- *Potential*: Reference Temperature in eV

  -Definition: T_{ref} -Value: 2 Volts

- *Velocity*: Thermal Temperature

  -Definition: $v_{ref} = \sqrt{\frac{kT_{ref}}{m_{ref}}}$ -Value: 1212.303955 m/s

3.2.2 Quasi-Neutrality

In Chapter 2, it was shown that the Debye length in the plume of a Hall Thruster is very small throughout the plume region. As a result, the plume can be regarded as everywhere quasi-neutral except in a small sheath region near solid surfaces. The a priori assumption of quasi-neutral has been used to greatly improve the performance of the PIC-DSMC simulation over conventional PIC models. This section discusses the computational benefits and physical limitations of the assumption and describes how the potential can be calculate in a PIC simulation without solving Poisson's equation.

In conventional PIC simulations (as outlined in Figure 1.7), the charge carried by the macroparticles is weighted to a computational grid to determine the charge density as a function of position in the plasma. The potential is then calculated by using an iterative scheme to solve Poisson's equation

$$\nabla^2\phi(x,y,z) = \frac{-\rho(x,y,z)}{\varepsilon_o} \qquad (3.2\text{-}1)$$

Typical solution schemes include ADI and multigrid iteration schemes. The size of the computational grid is limited by the need to capture the physics of particle interaction in the plasma. The Debye length is a critical quantity because it represents the length scale over which the plasma will shield out charge perturbations. In general, the cell spacing must be no more than twice the Debye length to properly capture the close range particle interactions. Though some researchers have used grid cells of larger sizes [*Samanta Roy* 1995], the Debye length is still considered a fundamental upper bound on the size of grid cells in a PIC simulation.

Sheaths, double layers, electron plasma waves, and other phenomena which occur on Debye length scales are non-quasi-neutral phenomena which are not generally of interest in Hall Thrusters plumes. Sheaths occur near solid surfaces, electron plasma waves can be time-averaged, and double layers are not known to occur with Hall thrusters. Assuming that the plume is quasi-neutral allows one to avoid Debye length limitations on cell spacing and solve for the potential without solving Poisson's equation. Quasi-neutrality also allows the use of fewer cells which leads to smaller and faster simulations. The main disadvantage of the quasi-neutral approach is that it is invalid near solid boundaries and in regions where the Debye length is of the same order as the length scale of the body. This can occur in the wake regions behind solar arrays and other bodies, but will not generally occur in regions directly impacted by the plume. Solid boundaries require the use of special boundary conditions which are discussed in Section 3.3.2. Overall, the computational benefits more than justify the limitations imposed by requiring the plasma to be quasi-neutral in the domain. This approach has therefore been used to simplify and speed up the PIC-DSMC plume model.

When quasi-neutrality is imposed on a plasma, Poisson's equation can no longer be used to calculate the potential distribution. This is because the net charge density is zero everywhere (to zeroth order), so the right hand side of equation (3.2-1) becomes zero, resulting in an incorrect solution. The quasi-neutral PIC-DSMC model is based on a hybrid PIC formulation, so the potential can be calculated by inverting the electron momentum equation. In a hybrid formulation, particles are used to represent ions and neutrals and a fluid formulation is used to represent electrons. The advantage of the hybrid approach is that one can avoid some of the time and length scale limitations imposed by electrons on the simulation. The disadvantage is that one loses the ability to track the details of electrons transport and that the hybrid approach assumes that the electrons have a Maxwellian distribution. In this case, we are more interested in the detailed motion of ions than of the electrons, so we use a hybrid approach. The general electron momentum equation is given by

$$m_e n_e (\vec{u}_e \cdot \nabla)\vec{u}_e = n_e e\left(\vec{E} + \vec{u}_e \times \vec{B}\right) - \nabla \cdot \vec{P} + m_e n_e \nu(\vec{u}_e - \vec{u}_o) \qquad (3.2\text{-}2)$$

Where u_0 is the drift velocity of a species that the electrons are interacting with via short range collisions. Weighting the charge carried by the macroparticles to the grid gives the ion density as a function of position in the plasma. Since the plasma is quasi-neutral, the electron density can be set equal to the ion density. In addition, the electron temperature and magnetic field are known quantities, so equation (3.2-2) can be combined with Maxwell's equations and inverted to obtain the pressure, velocity, and potential of the electrons in the plasma. Solving the resulting system is a non-trivial process and requires numerical methods traditionally used to solve the Euler and

Navier-Stokes equations. In the case of a Hall Thruster, however, a series of simplifying assumptions can be made. The five terms in equation (3.2-2) represent, from left to right, the inertial, electrical, magnetic, pressure, and collisional contributions to the electron momentum equation. Non-dimensionalizing equation (3.2-2) results in the following representations for each term.

$$1 = \frac{eEL}{m_e u_e^2} \& \frac{eBL}{m_e u_e} \& \frac{P}{\rho u_e^2} \& \frac{\nu L}{u_e} \qquad (3.2\text{-}3)$$

In Chapter 2 it was shown that the magnetic field can be ignored at distances greater than 25 cm from the thruster exit. We choose to ignore the magnetic field throughout the plume and disregard the magnetic term in equation (3.2-2), thus ignoring any Hall currents present in the plume region. The remaining quantities represent the relative contribution of each term to the momentum equation. The inertial term is of order one and its magnitude relative to the other terms depends on the electron drift velocity. The operating potential of the thruster is 300 V, but the voltage drop seen in the plume is expected to be much less. We approximate it as 100 V. The electrical term is then given by

$$\frac{eEL}{m_e u_e^2} = \frac{eV}{m_e u_e^2} \doteq \frac{1.777\text{x}10^{13}}{u_e^2}$$

For the pressure term, we assume a Maxwellian plasma in which the pressure tensor can be reduced to a scalar and treated using the expression

$$P = n_e k T_e$$

The electron temperature in the plume is approximately constant in the plume and is about 2 eV (see Figure 2.4). The pressure term then reduces to

$$\frac{P}{\rho u_e^2} = \frac{kT_e}{m_e u_e^2} = \frac{3.56 \times 10^{11}}{u_e^2}$$

The collision term depends largely on the collision frequency. The mean free paths shown in Table 2.2 show that electron-ion Coulomb collisions dominate over electron-neutral collisions. The momentum transfer cross section was calculated in Section 2.2.3 as 1.41 x 10^{-17} m^2. This cross section is based on the electron temperature and is valid as long as the electron drift velocity is less than the electron thermal velocity. The electron thermal velocity was calculated in section 2.2.2 as 9.52 x 10^5 m/s. The electron-ion collision frequency is given by the expression

$$\nu = n_i \sigma c_r$$

Where n_i is the ion number density. For c_r we again use the electron thermal velocity, which results in the following expression for the collision term

$$\frac{\nu L}{u_e} = \frac{1.34 \times 10^6}{u_e}$$

The magnitude of the terms in equation (3.2-3) is calculated as a function of velocity and plotted in Figure 3.2. The electron sonic speed is the same magnitude as the electron thermal speed, or about 9×10^5 m/s. Figure 3.2 shows that the pressure and electrical terms dominate the electron momentum equation at low drift velocities, though the collision term becomes important as the electron drift velocity approaches sonic speeds. The electrons in the plume are not believed to travel at supersonic velocities, so the dominant terms in the plume of a Hall thruster are the inertial and pressure terms. Equation (3.2-2) can therefore be simplified into the following form

$$n_e q\vec{E} - \nabla p = 0$$

which, for an isothermal Maxwellian distribution, can be written as

$$n_e = n_o \exp\{-e\phi / kT_e\} \qquad (3.2\text{-}4)$$

Equation (3.2-4) is the Boltzmann relationship and is identical to the "Barometric Law" given in Equation (1.1-2). This expression is the one used in our computational model. It is an entirely local formulation and requires no finite-difference approximation.

The Barometric law has been shown experimentally to apply to ion thruster plumes, so it is natural that it would also apply to Hall Thruster plumes. The electron number density can be determined from the distribution of charges particles in the simulation. Once the number density has been determined, the potential can be determined simply by inverting equation (3.2-4). Based on experimental measurements of the electron temperature, a constant temperature of 2 eV is used in our plume simulations. Because (3.2-4) contains no differential terms, the potential can be determined locally without use of an iterative solving scheme. This makes the formulation extremely efficient and easy to implement. The quasi-neutral PIC formulation has several major computational advantages over conventional PIC methods. First, because the formulation is quasi-neutral, the grid cell sizes are no longer limited to Debye length scales. Instead, the grid can be sized geometrically so as to resolve the gradients and other features of interest in the plume. Second, the potential can be obtained by inverting the electron momentum equation which in this case reduces to the Boltzmann relationship. The Boltzmann relationship is a scalar function which can inverted without the use of iterative solvers. This makes the scheme very computationally efficient and easy to implement and debug. The main disadvantages associated with the quasi-neutral scheme are that it does not apply in highly rarefied regions, that analytic sheath models must be developed and applied as boundary conditions to the simulation, and that one can not use the scheme to track the electrons in the plume region. In addition, although the method is nominally time accurate, the particle tracking method places numerical limits on the highest frequencies which can be accurately simulated using a given time step and grid length scale. These limits are discussed in Section 3.4.1 below.

3.3 Domain and Boundary Conditions

3.3.1 Computational Grid

Embedded Mesh and Data Storage

The PIC and DSMC methods both use grids to discretize the computational domain. Our PIC-DSMC plume model is based on a single embedded grid. An embedded grid* is a type of unstructured grid that uses embedded Cartesian meshes to resolve important areas of the domain. An example of an embedded grid is shown in Figure 3.3. Embedded meshes are a compromise between the simplicity of Cartesian grids and the flexibility of fully unstructured grids. A Cartesian grid is computationally fast and very simple to implement, but can not be used to model realistic geometries or to resolve large variations in length scale. This is problematic because the fundamental length scale of the plasma, the Debye length, varies as the inverse square root of the density. A Cartesian grid fine enough to resolve the areas of high density provides too much resolution in areas of low density. This results in too few particles per cell and destroys the accuracy of the simulation. *Samanta Roy* [1995] used a Cartesian grid with variable grid spacing to study the plume of an ion thruster interacting with the surfaces of a spacecraft. His technique is useful for very simple geometries, but can not easily be extended to general spacecraft geometries.

A fully unstructured grid discretizes the computational domain into triangles (2-D) or hexahedral elements (3-D) which can be fitted to complex surfaces. This approach is commonly used in computational fluid dynamics and in DSMC simulations. The main disadvantage of this approach is the complexity of the grid itself. To date, only one effort has made use of unstructured grids with the PIC method. Peng [1991] used a fully unstructured mesh to simulate sputtering off of the acceleration grids of ion engines. No effort was made to extend this work into the plume region. There is also some computational overhead associated with the use of unstructured grids with particle simulations. Weighting a particle to a grid requires that one know which grid cell contains the particle (its cell coordinates). On a Cartesian grid with uniform spacing, a particle's cell coordinates are given by

$$i = \frac{r - r_o}{h} \qquad j = \frac{z - z_o}{h} \qquad (3.3\text{-}1).$$

where r and z are the particle's position, r_0 and z_0 are the coordinates of the origin, and i and j are the coordinates of the cell that contains the particle (see Figure 3.4). Determining a particle's cell coordinates on an unstructured mesh is much more complex. Each particle must be explicitly tracked to determine when it crosses a cell boundary and what cell it is leaving/entering as it

* The term "mesh" will be used to refer to a single level in an embedded mesh. The term "grid" will be used to refer to the collection of all the meshes in a domain.

crosses. Though the process is still relatively fast, because it occurs repeatedly during each timestep, it is quite time consuming compared to moving particles on a Cartesian grid. It is therefore desirable to use a simpler grid formulation to lower the computation time.

The position of a particle on an embedded mesh* can generally be determined using equation (3.3-1). Additional computations are required when particles jump between meshes, but since the perimeter of a mesh scales as O(N) while the number of nodes scales as O(N^2), the computational penalty is relatively small. In our simulation, embedded meshes are stored in a tree like structure as shown in Figure 3.5. Each fine mesh is a called a "child" of the coarse mesh one level above it. Each coarse mesh is a "parent" to its child meshes. A given mesh can have multiple children but only one parent. At the top of the tree is the coarsest mesh in the simulation. This coarse mesh defines the limits of the overall computational domain. When a particle crosses a mesh boundary, it is "jumped" to a parent or child mesh as appropriate. Its position on the new mesh is found using equation (3.3-1). An advantage to this storage scheme is that grid related functions can be carried out recursively. Once a single routine has been constructed to calculate the electric potential at each node on a given mesh,. the routine is first used to evaluate the top mesh and then calls itself to evaluate each child mesh. Each child mesh calls the routine again to evaluate its children, and so on Because the formulation is recursive, the simulation can be easily scaled to arbitrary numbers of embedded meshes. In addition, a routine verified on one parent-child combination will generally work without modification on any further meshes.

A relatively small amount of information is needed to define an embedded mesh. This information includes the position of the local origin relative to parent and child meshes, the cell spacing and the number of nodes in each direction. We also store information about the boundaries of each cell and its relationship to parent and child meshes. The grid occupies more memory than a traditional embedded mesh, but the additional information lowers the computation time by facilitating the tracking of particles as they move through the domain. The basic grid information is stored in two different data structures, one associated with nodes and one associated with cells. These data structures are defined using the following C code.

* The term "mesh" will be used to refer to a single level in an embedded mesh. The term "grid" will be used to refer to the collection of all the meshes in a domain.

```
typedef struct {
  char celltype;             /* Switch to tell if cell borders on fine cells */
  edgedata edge[4];                        /* One for each of the four edges */
  float local_time;                     /* Local time count (for MCC module) */
  float sigma_v_max;                /* Max sigma * velocity (for MCC module) */
  int ncollision;                       /* Number of collisions in that cell */
  int summed_ncollision; /* Sum to calc. average number of collision in cell */
  unsigned int *npart;                    /* Number of particles in the cell */
  particledata **part;               /* Pointer to list of particles in cell */
} celldata;


typedef struct {
  float weight;                 /* Weight factor for density multiplication */
  float ex;                                /* Electric field in x direction */
  float ey;                                /* Electric field in y direction */
  float ndensity;                                 /* Neutral number density */
  float phi;                                                 /* Potential */
  float rhominus;                                /* Negative charge density */
  float summed_vx;                             /* Sum used to calc. ave. vx */
  float summed_vy;                             /* Sum used to calc. ave. vy */
  float nvsamples;                    /* Running "sum" for velocity samples */
  float summed_ndensity;          /* Sum to calc. time ave. neutral density */
  float summed_rho; /* Sum used to calculate time ave. (ion) charge density */
  float summed_phi;            /* Sum used to calculate time ave. potential */
  float sndensity[5];
  float species_vx[5];
  float species_vy[5];
  float species_samples[5];
  char switchns;      /* Switch to indicate which way to difference e field */
  char switchew;
  short int nobj;    /* Index to object node is assocaited with */
  short int n;       /* Index to node on the object */
} nodedata;
```

The code shown above defines two data structures: one for cells (named “celldata”) and one for nodes (named “nodedata”). A set of data is stored for every cell or node in the domain. The next two sections briefly discuss the contents of each of these structures. Several of the parameters used in these structures are flags which are defined using arbitrary constants . A flag may be designated by the values TRUE and FALSE, for instance, which are traditionally represented by the values one and zero respectively. The actual value of these constants is not significant and are omitted from the descriptions given below.

Cell Data Structure

celltype is a flag indicating whether a cell borders on a child mesh. It has two possible values.

- DEFAULT_CELLTYPE: the cell does not border on a child mesh
- BORDERS_ON_FINE: the cell does border on a child mesh

edgedata is a set of four data structures, one for each of the four faces of the cell. These faces are referred to using the names NORTH, SOUTH, EAST and WEST as shown in Figure 3.4. Each of these structures contains two components. *type* is a flag indicating the type of the edge. It can take on the following values.

- INTERIOR_BND: the edge lies on an interior boundary of the domain (typically a solid surface). Particles crossing this edge are subject to surface boundary conditions.
- EXTERIOR_BND: the edge lies on an exterior boundary of the domain. Particles crossing this edge are subject to exterior boundary conditions.
- SAME_CELL: the edge borders on a cell with the same resolution.
- FINE_CELL: the edge corresponds to the edge of a child mesh. The bordering cells are of finer resolution.
- COARSE_CELL: the edge corresponds to the edge of a parent mesh. The bordering cells are of coarser resolution.
- NO_BND: the edge borders on nothing. This is used in the interiors of solid objects.

The second component is the *index* component. It contains an integer value pointing at a grid or an object as follows.

- If the edge is an INTERIOR_BND, it designates the object lying on that boundary.
- If the edge is a FINE_CELL it designates a child mesh connected to the boundary.

local_time and *sigma_v_max* are values used to do local time counting and selection-rejection for collisions (see section 3.5.1). *ncollision* is the number of collisions which occurred in the cell during the most recent timestep. *summed_ncollisions* is the number of collisions which have occurred during the entire simulation. *npart* is an array of numbers recording the number of particles of each species (neutral, ion, or double ion) presently located inside the cell. Finally, *part* is an array of points to linked lists of the particles presently located in the cell. The linked list memory structure is described in the computational notes in section 3.7.

Node Data Structure

Much of the information contained in the node data structure corresponds to physical values used to move particles and to produce user output.. *ex* and *ey*, for instance, correspond to the two components of the electric field. Similarly, *vx* and *vy* correspond to two components of drift velocity, and *ndensity, phi, rhominus* correspond to physical quantities as described in the comments included in the definition. The rest of the structure contains information about the structure of the grid itself. This information includes:

weight is the weighting factor used to calculate the particle number density for the node. These weighting functions are described below.

switchns is a flag indicating the type of finite-difference formulation used to calculate the axial electric field at this particular node. These formulations are discussed in section 3.4.2. The flag can take the following values.

- CENTER indicates that the center difference formulation should be used.

- NO_DIFF indicates that the electric field should not be calculated. This is used in the interiors of solid objects.
- NORTH indicates that a forward difference formulation should be used.
- SOUTH indicates that a backward difference formulation should be used.

Similarly, *switchew* indicates the finite-difference formulation used to calculate the radial electric field. The flag can take on the following values.

- CENTER indicates that the center difference formulation should be used.
- NO_DIFF indicates that the electric field should not be calculated. This is used in the interiors of solid objects.
- EAST indicates that a forward difference formulation should be used.
- WEST indicates that a backward difference formulation should be used.

Finally, if a node is located at an object boundary, two other parameters are set during grid generation. *nobj* designates the object that the node is associated with and *n* designates a node on the object that corresponds to the node on the grid.

Additional Notes

As was stated earlier, embedded meshes can be used to resolve areas of different density in a particle simulation. The proper use of embedded grids is a bit tricky with particle based simulations. In CFD applications, users can almost arbitrarily increase the grid resolution in order to increase the accuracy of the simulation. The only tradeoff is in the computation time, which increases linearly with the number of nodes in the simulation In a PIC-DSMC code, however, increasing the grid resolution without increasing the number of particles results in a decrease in the number of particles in each cell. This actually decreases the local accuracy by increasing the noise in the simulation. However, smaller cells are necessary and appropriate to resolve areas of high gradients and high densities. High density areas naturally contain more particles, so the overall number of particles per cell can be held constant throughout the simulation while resolving areas of interest. Experience with our simulation suggests that an efficient procedure is to use one level of grid to resolve a small area just outside the thruster exit and to avoid using embedded meshes elsewhere in the domain. In axisymmetric simulations, the macro particle weights are generally adjusted so that there are at least 10 particles per cell within the embedded mesh and throughout the rest of the domain.

Another point which should be noted is that we use the same computational mesh for both the PIC and DSMC portions of the simulation. In principle, it is possible to use separate meshes to resolve the two difference processes: one for the DSMC mesh and one for the PIC mesh. However, some difficulties may arise if this method is used. In general, the mean free path in a plasma is greater than the Debye length (*Bittencourt* Ch. 1). The DSMC grid would therefore use larger cells than the PIC grid, which allows particles which are electrically isolated from each other to interact through collisions. As a result, in a non-quasi-neutral formulation, particles inside a sheath could collide with particles far away from the sheath (in an electrical sense), creating non-physical interactions. To avoid such problems, it seems wiser to use a single high resolution grid for both simulations even though this incurs a computational penalty in the DSMC portions of the simulation. More work is needed to clarify the computational and physical issues surrounding the use of separate grids for a PIC-DSMC simulation.

3.3.2 Cell Weighting Factors

During the PIC portion of the simulation, the local charge density is determined by weighting particles to the grid and then multiplying by a cell weighting factor. At most nodes, the charge density can be determined by dividing the charge collected at the node by the volume of the cell, i.e.

$$\rho_{i,j} = \frac{q_{i,j}}{h^2} \qquad (3.3\text{-}2)$$

where $q_{i,j}$ is the charge collected at that particular node and h is the cell spacing of the local mesh. However, since the charge carried by each particle is shared among the four nodes of the cell that contains it, nodes along surfaces and at corners naturally collect less charge than their freestanding counterparts. This occurs because boundary nodes collect charge from a smaller area than their freestanding equivalents, as shown in Figure 3.6. Before applying equation (3.3-2), it is necessary to multiply the charge collected by the ratio of the cell's total area divided by the node's effective area, as shown in Figure 3.6. This ratio is referred to as the node weighting factor. For most nodes, the weighting factor is one. Particles on boundaries generally have weighting factors of 2 except at corners, where different factors apply. Weighting factors are not required at embedded mesh boundaries. Particles are recursively weighted to parent meshes and, along mesh boundaries, to child meshes so charges are weighted from the entire area around the node (as shown in Figure 3.6). This helps prevent the formation of discontinuities along the edges of embedded meshes.

3.3.3 Exterior Boundary Conditions

Computational boundaries occur at the edges of the domain and at solid surfaces (such as the surface of a solar array). Particles crossing these boundaries are subject to boundary conditions. This section describes the boundary conditions imposed at the exterior edges of the computational domain. Boundary conditions imposed on solid surfaces are described in section 3.6. In conventional PIC simulations, boundary conditions are imposed on the potential and on particles interacting with computational boundaries. The quasi-neutral potential formulation used in this simulation is an entirely local formulation that requires no explicit boundary conditions. In many cases, however, setting a particle boundary conditions implicitly sets a potential boundary condition as well though the Boltzmann relationship.

Boundary conditions are checked during the particle moving phase of the simulation. As discussed in section 3.3.1, each cell contains a series of flags indicating the state of each of the cell's faces. A cell north face, for instance, is designated as an interior boundary, exterior boundary, cell-to-cell boundary, etc. When a particle crosses a cell boundary, the simulation checks the appropriate flag to see if the particle has crossed an exterior boundary. If it has crossed such a boundary, the particle will either be absorbed or reflected as is appropriate. The r = 0 boundary (West) is an axis of symmetry. Since the axis has zero width, no particles should strike this boundary. In practice, however, the use of a finite timestep allows some particles to strike the boundary. These particles are reflected back into the domain by reversing the component of the velocity normal to the boundary and reflecting the particle's initial position across the line of symmetry, as shown in Figure 3.7. When the particle's new position is calculated at the end of the time step, it looks as though the particle came from the other side of the boundary. This accurately represents the presence of an axis of symmetric. The particle also looks as though it struck a solid boundary and bounced off it; the two cases are microscopically indistinguishable. The other exterior boundaries are absorbing boundaries representing vacuum or a background gas. Particles striking an absorbing boundary are immediately deleted from the simulation.

If a background gas is present in the simulation, new particles must be loaded at the edges of the domain to simulate the flux of particles entering from the background due to thermal motion. The magnitude of the particle flux is the Maxwellian thermal flux, i.e.

$$\Gamma = \frac{n\bar{c}}{4} = \frac{n}{4}\sqrt{\frac{8kT}{\pi m}} \qquad (3.3\text{-}3)$$

The velocity distribution of particles entering the domain is based on a Maxwellian distribution. Each new particle requires three velocity components. Two of these components are parallel to the boundary. The velocity in these directions is set using a Gaussian distribution produced using equation (3.3-8). The third component of the velocity is normal to the boundary. The velocity in this direction is set using a "half-Maxwellian" distribution function. A half-Maxwellian is not simply half of a Gaussian distribution because particles moving at different velocities have different probabilities of crossing the boundary into the domain. The half-Maxwellian is derived from the Gaussian as follows.

The basic 1-D Gaussian distribution function is given by

$$f(v_x) = \frac{1}{\sqrt{2\pi}v_t}\exp\left(\frac{(v_x - v_d)^2}{2v_t^2}\right)$$

Where v_t is the thermal velocity (as defined in equation 3.3-9) and v_d is the drift velocity. $f(v_x)$ gives the fraction of all molecules with velocities between v_x and $v_x + dv_x$. An exterior boundary represents a planar cut though the background gas, as shown in Figure 3.8. The number of particles in a given velocity range v_x+dv_x that will cross the boundary in a time dt is given by the fraction of all particles with velocities between v_x and dv_x contained in a rectangular volume of length $v_x dt$, as shown in Figure 3.8. The flux of particles crossing the boundary at a given velocity is therefore given by

$$I(v_x) = \frac{n v_x}{\sqrt{2\pi} v_t} \exp\left(\frac{(v_x - v_d)^2}{2v_t^2}\right)$$

Integrating $I(v_x)$ over all positive velocities gives the total flux of particles across the surface. This integration can be carried out by defining $q = (v_x - v_d)$, dividing up the integrand into two terms, and integrating by parts. The result is

$$\Gamma = n\left\{\left(\frac{v_t}{\sqrt{2\pi}}\right)\exp\left(-{v_d^2}\Big/{2v_t^2}\right) + \frac{v_d}{2}\left[1 + \operatorname{erf}\left(\frac{v_d}{\sqrt{2}v_t}\right)\right]\right\}$$

When the drift velocity is zero, this expression reduces to (3.3-3). The quantity of direct interest to the simulation is the cumulative distribution function (CDF) for the velocity of particles crossing the boundary. The probability that a particle crossing the boundary has a velocity less than y is given by

$$P = \frac{1}{\Gamma}\int_0^y \frac{n}{\sqrt{2\pi} v_t} v_x \exp\left(-{(v_x - v_d)^2}\Big/{2v_t^2}\right) dv$$

This expression can also be evaluating using the substitution $q = (v_x - v_d)$ and gives the following result.

$$P = \frac{n v_t}{\sqrt{2\pi}\Gamma}\left[\exp\left(-{v_d^2}\Big/{2v_t^2}\right) - \exp\left(-{(y - v_d)^2}\Big/{2v_t^2}\right)\right] + \frac{n v_d}{2\Gamma}\left[\operatorname{erf}\left(\frac{v_d}{\sqrt{2}v_t}\right) \mp \operatorname{erf}\left(\frac{\mp(y - v_d)}{\sqrt{2}v_t}\right)\right] \quad (3.3\text{-}4)$$

Where two signs are present, minus should be used when $y < v_d$ and plus should be used when $y > v_d$. The drift velocity is specified at each boundary. For a stationary background gas, the drift velocity is always zero.

The flow from an exterior boundary is evaluated using the following procedure. The number of particles to enter the domain in a given timestep is determined from equation (3.3-3). The particle is then assigned velocity components in the two directions parallel to the exit plane using expression (3.3-9). The velocity component normal to the boundary is calculated by choosing a random number, P, from a uniform distribution between 0 and 1 and then inverting equation (3.3-4) to solve for the velocity of the particle, y. When repeated over many particles, the result is a "half-Maxwellian" distribution. Equation (3.3-4) can not generally be inverted analytically, so it is inverted computationally using a lookup table. The position of the particle is also determined randomly. The components of position are chosen so the particle has an equal chance of being anywhere within a rectangular region adjacent to the boundary with thickness $v_x\Delta t$. This correctly simulates the velocity and position of particles entering the domain from the body of a background Maxwellian gas.

3.3.3 Source Modeling

Particles are loaded into the simulation at each time step to simulate the exit flow from an SPT-100. A plasma source model has been developed to simulate the plasma and neutral flow from an SPT-100 thruster. The model is highly empirical and is largely based on fits to experimental data and reasonable guesses about the conditions at the anode exit. This section describes the SPT-100 source model and its experimental basis and describes the assumptions made to construct a source model.

As was discussed in Chapter 2, the SPT-100 thruster releases neutrals, ions, and electrons into the plume region. Since electrons are simulated using the Boltzmann equation, the flow of electrons from the thruster is not directly simulated in the plume model. The flux of ions and neutrals from the thruster is given by equation (2.1). The number of macroparticles entering the domain in each time step is equal to the flux multiplied by the time step and divided by the appropriate macroparticle weight. The macroparticles are released from a simulated annular exit with dimensions identical to those of the acceleration channel. In the axisymmetric simulation, this orifice is placed so the domain's axis of symmetry is aligned with the thrust vector. Because the cathode can not be directly simulated in an axisymmetric geometry, the flow from the cathode is diverted into the acceleration channel and released through the same annular exit as the flow from the anode. This approximation is not used in three dimensional simulations.

The distribution of ions leaving the anode is determined using an empirical model developed from experimental measurements of the ion current density 4 mm from the thruster exit. These measurements give the magnitude and direction of the ion current as a function of radial position at an unspecified propellant flow rate and are shown in Figure 3.9 [*Gavryushin* 1981]. The exact type of thruster is not specified in the paper, though it appears to be an early SPT-70 or SPT-100 thruster. The measurements were taken using planar probes, but no error bars are included. The laboratory data shows that the ion current density and its direction are a strong function of radius. The beam divergence angle becomes as large as 50 degrees at the edges of the acceleration channel. The magnitude and divergence angle of the ion current was taken from Figure 3.9 and fitted using high order polynomial functions. The results are

$$\alpha = 1730 - 2.30\times10^5 r + 1.06\times10^7 r^2 - 2.05\times10^8 r^3 + 1.45\times10^9 r^4 \qquad (3.3\text{-}5)$$

$$j = -1210 + 8.40\times10^4 r - 1.78\times10^6 r^2 + 1.18\times10^7 r^3 \qquad (3.3\text{-}6)$$

Where α is the local divergence angle in degrees (as measured from the centerline, with negative towards the centerline), j is the current density in mA/cm^2 and r is the radial position in meters.

Additional assumptions about the flow are required to reconstruct the distribution function at the exit plane. The first assumption is that ions leave the thruster with a drift velocity of 17020 m/s in the r/z plane. This value corresponds to the measured specific impulse of an SPT-100 after excluding the flow to the cathode [*Manzella et al.* 1995]. The value is probably slightly low since the plume divergence angle lowers the effective I_{sp} of the SPT-100 thruster. When this assumption is combined with equation (3.3-6), variations in current density correspond to variations in number density. Integrating (3.3-6) gives the total ion flow from the thruster. This quantity can be used to normalize equation (3.3-6), resulting in a probability distribution function. Integrating that result gives the following cumulative distribution function.

$$P = 2.55 - 1.67\times10^{4} r^{2} + 7.71\times10^{5} r^{3} - 1.23\times10^{7} r^{4} + 6.50\times10^{7} r^{5} \qquad (3.3\text{-}7)$$

Equation (3.3-7) gives the probability that an ion crossing the exit plane has a radial position less than r. When r is equal to the inner anode radius, P = 0. When r is equal to the outer anode radius, P = 1. Equation (3.3-7) can be used to produce a distribution of particles corresponding to the distribution shown in Figure 3.9. P is set to a random number between 0 and 1 and (3.3-7) is inverted to give the position at which the ion should be released relative to the thruster's centerline. When this process is done repeatedly, the result is the ion current density distribution represented by equation (3.3-6). Equation (3.3-7) is inverted computationally using a lookup table. This was found to be the fastest implementation of the solution.

Once the particle's radial position is known, Equation (3.3-5) is used to determine the particle's divergence angle relative to the centerline. Since the particle's total velocity is fixed at 17020 m/s, the divergence angle can be used to calculate the axial and radial drift velocities of the particle. The result is the mean axial and radial velocity of particles at a given radial position. In order to fill out the distribution function, thermal velocity components must be added in the axial and radial directions. We use Gaussian distributions based on estimates of the ion temperature at the anode exit. Gaussian distributions are constructed using an algorithm presented in *Birdsall* [pg. 391]. A Maxwellian distribution characterized by the thermal velocity v_t is constructed using the following expression.

$$v = v_t\left(\sum_{i=1}^{M}\beta - \frac{M}{2}\right)\left(\frac{M}{12}\right)^{-1/2} \qquad (3.3\text{-}8)$$

This expression creates a Gaussian from the sum of M randomly chosen numbers between 0 and 1. v_t is the thermal velocity and is given by

$$v_t = \sqrt{\frac{kT}{m}} \qquad (3.3\text{-}9)$$

where T is the temperature of the distribution and m is the mass of the ion/neutral particle. We use M = 12, which gives an excellent approximation of a Maxwellian distribution. The result is added to the axial and radial velocities produced by equations (3.3-5) and (3.3-6) to give the final particle velocity.

The radial and axial temperatures are treated independently. The axial ion temperature is a measure of the scatter that appears because ions created in different parts of the acceleration channel fall though different acceleration potentials. As discussed in Chapter 1, *Manzella* [1994] used laser induced fluorescence to measure the axial ion temperature and reported a temperature of 3.4 eV. *Myers and Manzella* [1994], *Absalamov* [1992] and *Gallimore* [Personal Communications 1996] used retarding potential analyzers to measure the ion energy distribution and produced results which are roughly consistent with this ion temperature. Sample measurements from Absalamov are shown in figure 2.2. The apparent spread in the ion energy distribution is deceiving. Although the peak is almost 50 volts wide, the actual temperature is much less than this because this plot show the energy rather than the velocity distribution. The actual temperature distribution is derived in section 4.2.4. We examined the effect of varying the axial ion temperature on the simulation and used temperatures ranging from 3.4 eV to 34 eV. These results are discussed in section 4.2.4. The temperature of the ions in the radial direction has not been measured. We assume a radial temperature of 8000 K in the plume simulation.

Ions emerging from the thruster also have non-zero azimuthal velocities. The azimuthal velocity of ions emerging from an SPT-100 thruster was measured by *Manzella* [1994] using laser induced fluorescence. He measured azimuthal drift velocities of 250 m/s and an azimuthal temperature of 800 K. These values are also used in the plume simulation.

The neutral distribution in the near field region of the acceleration channel has not been measured. The simulation assumes that neutrals leaving the thruster have a temperature of 1000 K (about 0.1 eV) and are choked at the channel exit. The radial and azimuthal drift velocities are assumed to be zero and the axial drift velocity is given by the sonic velocity, i.e.

$$a = \sqrt{\frac{\gamma k T}{m}}$$

Thermal velocities corresponding to a temperature of 1000 K are calculated using expression (3.3-8) and are added in all three directions.

3.4 Particle-in-Cell Theory and Techniques

3.4.1 Particle Motion

The Hall Thruster simulation moves both ions and neutrals by integrating the particle equations of motion using the leapfrog method. In Chapter 2, it was established that the magnetic field in the plume would be ignored in this simulation. Since collisions are carried out in a separate step, in axisymmetric coordinates, the equations of motion are simply

$$\frac{dv_r}{dt} = \frac{qE_r}{m} + \frac{v_\theta^2}{r} \qquad \frac{dv_z}{dt} = \frac{qE_z}{m} \qquad \frac{dv_\theta}{dt} = -\frac{v_r v_\theta}{r} \tag{3.4-1}$$

$$\frac{dr}{dt} = v_r \qquad \frac{dz}{dt} = v_z$$

where q/m is the charge ratio of the macroparticle. For neutrals, this ratio is zero. The simulation is axisymmetric with swirl, so the simulation tracks three velocity components and two position components. The electric field in the azimuthal direction is zero by symmetry. System (3.4-1) is integrated using the leapfrog method, which is a well known second order accurate method used in most particle based simulations. The leapfrog method is discussed in detail in several books including *Birdsall* [pp. 12-15 and pp. 184-196]. System (3.4-1) is replaced by finite difference equations of the form

$$v_r^{t+1} = v_r^t + \left(\frac{qE_r}{m} + \frac{v_\theta^2}{r}\right)\Delta t \qquad v_z^{t+1} = v_z^t + \frac{qE_z}{m}\Delta t \qquad v_\theta^{t+1} = v_\theta^t - \frac{v_r v_\theta}{r}\Delta t$$

$$r^{t+1} = r^t + v_r \Delta t \qquad z^{t+1} = z^t + v_z \Delta t$$

The velocity and position equations are evaluated at different points in the timestep, as shown in Figure 3.10. The simulation advances x and v based on each other even though they are not actually known at the same time. Strictly speaking, the initial position and velocity of the particle should be offset from each other by half a timestep. Since our velocities and positions are chosen randomly, no explicit offset is used in our simulation. As discussed in *Birdsall*, the model is second order accurate and very simple to implement. The method produces varying errors depending on the frequency of the process simulated. In general, a harmonic oscillator of radian frequency ω will be simulated with no amplitude error for $\omega\Delta\tau < 0.2$ and the phase will advance as (*Birdsall* pg. 14)

$$\omega\Delta t + \frac{1}{24}(\omega\Delta t)^3 + \text{H.O.T.}$$

It has been noted that the leapfrog method will simulate waves for some "tens of cycles" with "acceptable" accuracy. This criterion sets an upper limit on the frequencies which can be accurately simulated using the quasi-neutral PIC-DSMC plume model.

3.4.2 Electric Field

In order to calculate the electric field, it is necessary to know the ion charge density throughout the computational domain (in principle, it is also necessary to know the electron density, but in this case, the plasma is quasi-neutral by assumption so $n_e = n_i$). The ion charge density is calculated by weighting the charged particles on the domain to the nodes of an embedded mesh using a first order area weighting method. Consider a particle inside an axisymmetric grid cell as shown in Figure 3.11. The charge Q carried by the particle P is divided among the four nodes of the cell based on the volume of the section of the cell located across from the node. The fraction of the charge weighted to node 1, for instance, is based on the volume of section V_1 divided by the total volume of the cell. This weighting in accomplished through the use of weighting factors. The radial weighting factors for an axisymmetric geometry were developed by *Ruytan* [1993] and are given by

$$S_j = \frac{\left(r_{j+1} - r\right)\left(2r_{j+1} + 3r_j - r\right)}{2\left(r_{j+1}^2 - r_j^2\right)} \qquad S_{j+1} = \frac{\left(r - r_j\right)\left(3r_{j+1} + 2r_j - r\right)}{2\left(r_{j+1}^2 - r_j^2\right)}$$

Where S is the weighting factor and subscripts refer to grid coordinates (see Figure 3.11). The axial weighting factors are simply Cartesian weighting factors given by

$$S_i = \frac{z_{i+1} - z_p}{z_{i+1} - z_i} \qquad S_{i+1} = \frac{z_p - z_i}{z_{i+1} - z_i}$$

The fraction of the total charge assigned to each node is given by products of the four weighting factors, as shown in Figure 3.11. These products are given by W_1, W_2, W_3 and W_4 respectively. The sum of W_1, W_2, W_3 and W_4 is always unity.

Once charges have been weighted to the grid, the charge density is determined by multiplying the collected charge at each node by a node weighting factor (discussed in Section 3.3.2) and dividing the result by the total volume of the cell. The potential is then determined directly using equation (3.2-4). The electric field is determined by differentiating the potential using a finite-difference formulation. We use a second order center difference formulation of the form

$$E^r_{i,j} = -\frac{\phi_{i,j+1} - \phi_{i,j-1}}{2h} \qquad E^z_{i,j} = -\frac{\phi_{i+1,j} - \phi_{i-1,j}}{2h} \tag{3.4-2}$$

Physically, a center difference is the preferred finite difference approximation. However, Equation (3.4-2) can not be used along mesh and domain boundaries because the leading or trailing node is undefined at those locations. The electric field is then determined using a second order forward or backwards difference approximations of the form

$$\text{Forward: } E^z_{i,j} = -\frac{-3\phi_{i,j} + 4\phi_{i+1,j} - \phi_{i+2,j}}{2h}$$

$$\text{Backward: } E^z_{i,j} = -\frac{3\phi_{i,j} - 4\phi_{i-1,j} + \phi_{i-2,j}}{2h} \tag{3.4-3}$$

Similar formulations are used for the radial electric field. One sided difference approximations are used along domain edges, along solid surfaces, and along the edges of embedded meshes. The use of one sided difference approximations along the edges of embedded meshes could introduce grid artifacts to the electric field, but in practice, no discontinuities have been observed in any of the simulations. It should be noted that no independent boundary conditions are imposed on the potential. Because equation (3.2-4) is a local formulation, potential boundary conditions are not required in this quasi-neutral simulation.

3.5 DSMC Theory and Techniques

3.5.1 Multi-Species Time Counter and Selection-Rejection

CEX collisions are modeled between move steps using a modified multi-species DSMC method which is outlined in Figure 3.12. The method used is a multi-species selection-rejection scheme which uses a single global time counter to determine collision rates. It is based closely on *Elgin*, which is in turn based on methods described by *Bird* [1976]. The method supports particles of different species and different macroparticle weights, though the method described here is restricted to a single weight for each type of particle (neutral, ion, or double ion). A well known and fairly conservative variation of the DSMC method is used in this work. More computationally efficient and advanced methods have been described in recent work including the most recent edition of *Bird* [1994]. Well known methods were used in order to simplify the collision module. Better computational performance may be achievable using more advanced variations of the DSMC method.

The first step in a traditional DSMC program is to sort the particles by their location and create lists of the particles present in each cell. Our simulation tracks particles as they cross from cell to cell during the move phase, so no separate sorting stage is necessary. Each particle contains a record of the mesh and cell that contains it, and each cell contains lists of each type of particle (ion, double ion, or neutral) present in the cell. Each cell has a single local time counter which is set equal to zero at the beginning of the simulation. The basic procedure is the same that described in Section 1.1.4. Each iteration, the program begins by adding one timestep to the local time counter. The program then compares the value of a cell's local time counter to the global time counter. If the local time is greater than the global time, the simulation proceeds to the next cell. If the local time is less than the global time, a pair of collision partners is selected randomly based on the selection-rejection method. When a collision takes place, the local time counter is incremented based on an inverted collision frequency. The local time counter is compared to the global time counter, and the process is repeated until the local time exceeds the global time.

The selection-rejection and local time counter have been modified to account for multiple species undergoing multiple collisions. A total of three different species are included in the axisymmetric plume simulation: Xenon neutrals, Xenon ions, and Xenon double ions. Xenon beam ions and Xenon CEX ions are labeled differently in the simulation (i.e. they show up as different colors when displayed using visualization software), but are treated the same in the collision process. This is also the case for Xenon double ions. Table 3.1 shows the collision processes included in the present simulation.

CEX	Elastic
Xe-Xe^+ Xe-Xe^{++}	Xe-Xe Xe-Xe^+ Xe-Xe^{++}

Table 3.1: Collisions Simulated in Plume Simulation

For completeness, elastic collisions are simulated even though work presented in Chapter 2 indicates that they are unlikely to affect the physics of Hall thruster plumes. Additional collision types could also be included in principle. Notably missing are Xe^+-Xe^+ collisions due to Coulomb interactions. Simulating these collisions is computationally impractical for reasons discussed at the end of section 3.5.1. Omitting these collisions has little effect since the mean free path for ion-ion collisions is long throughout the plume region (as shown in Table 2.2). We note, however, that Coulomb collisions could become significant if the PIC-DSMC model were used to simulate plumes of lower ion temperature or higher ion density. In addition, for computational reasons, Xe-Xe collisions were also omitted from simulations conducted with neutral background gas. This also has little effect on the plume (see section 3.5.2).

A selection-rejection scheme is used to choose collision pairs and a single local time counter is used to determine the collision frequency for all collision processes. Both schemes have been modified to account for multiple collision species and variable macro-particle weights. The next two sections describe these limitations in detail and outline the computational limitations of each method. The details of the collision dynamics and calculation of the collision cross sections are presents in section 3.5.2.

Selection-Rejection

It is useful to begin with some nomenclature. When a collision occurs between two particles of unequal macroparticle weights, the higher weighting factor will be referred to as W_u ("upper weight") and the lower weighting factor will be referred to as W_l ("lower weight"). Selection-rejection for multi-species collisions is done by picking a "random" pair of particles and then applying the following criterion:

- Collision occurs if:

$$\beta < \frac{Q}{Q_{max}} \qquad Q = W_u \sigma_{ij} c_r \qquad (3.5\text{-}1)$$

Q_{max} is the largest Q which has been seen in this particular cell among all possible collisions, even those of different types or between difference species. A local value of Q_{max} is used because different parts of the domain may see significantly different values of Q. If a global counter were used instead, those parts of the domain in which $Q << Q_{max}$ would have small values of Q/Q_{max}, so the simulation would spend very large amounts of time rejecting otherwise valid collisions.

When a local Q_{max} is used, each cell will adjust itself to local conditions. Because the two particles have different weighting factors (and therefore represent different numbers and masses of particles), modifying the velocity and momentum of both particles in every collision would violate the overall conservation of momentum and energy. To ensure conservation of energy and momentum, another selection rejection is done based on the weights of the two particles, i.e.

$$\beta < \frac{W_l}{W_u} \qquad (3.5\text{-}2)$$

Where β is a random number chosen from a uniform distribution between 0 and 1. If equation (3.5-2) is true, the simulated collision is carried out as though the two particles had the same macroparticle weight. If equation (3.5-2) is false, the simulated collision is carried out as before, but changes are made only to the particle with the lower weight. The particle with the higher weight is allowed to continue on its original trajectory. In this scheme, energy and momentum are conserved at a statistical level even though they are not conserved in individual collisions. In the limit as the weights become equal, our scheme becomes a conventional DSMC method. In the other limit, as the neutral weight becomes infinitely large, our scheme approaches the conditions simulated in a PIC-MCC method. In all cases, all members of one species (either ions or neutrals) have the same macroparticle weight. Note that the selection criterion used in (3.5-1) is based on the W_u. This is because each computational collision represents W_l real collisions, so the selection probability must be increased by the factor W_u/W_l.

In order for (3.5-1) to work correctly, the process for "randomly" choosing particles within a cell must be weighted by the types of collisions the particle can undergo. In this case, since all of the collision pairs shown in Table (3.1) involve a neutral as the first collision partner, the first particle is randomly selected from the neutrals in the cell. The second collision partner can be an ion, double ion, or neutral, but since ions and double ions can participate in both elastic and CEX collisions, each ion/double ion is weighted by a factor of two with respect to the ions in the simulation. The code used to implement this selection scheme is shown below.

```
/* choose xe neutral, ion, or double ion for CEX or elastic collision */
    n2 = ran2(idum)*(2.0*localnpart[XE_ION]+ 2.0*localnpart[XE_2_ION]
                      +localnpart[XE_NEUTRAL]);
    /* Select a random particle */
    if (n2 < localnpart[XE_ION]) {
      collisiontype = SINGLE_CEX;
      part2 = *(local_part[XE_ION][n2]);
      goto end_select_n2;
    }
    else n2 -= localnpart[XE_ION];

    if (n2 < localnpart[XE_ION]) {
      collisiontype = XE_XE_ION_ELASTIC;
      part2 = *(local_part[XE_ION][n2]);
      goto end_select_n2;
    }
    else n2-= localnpart[XE_ION];

    if (n2 < localnpart[XE_2_ION]) {
      collisiontype = DOUBLE_CEX;
      part2 = *(local_part[XE_2_ION][n2]);
      goto end_select_n2;
    }
    else
      n2 -= localnpart[XE_2_ION];

    if (n2 < localnpart[XE_2_ION]) {
      collisiontype = XE_XE_DOUBLE_ION_ELASTIC;
      part2 = *(local_part[XE_2_ION][n2]);
      goto end_select_n2;
    }
    else
      n2 -= localnpart[XE_2_ION];

    if (n2 < localnpart[XE_NEUTRAL]) {
      /* Half the neutral-neutral choices have to be thrown out */
      /* Because the Xe-Xe collision rate is 0.5*n^2*c*sigma */
      if (ran2(idum) < 0.5) {
        collisiontype = XE_XE_ELASTIC;
        part2 = *(local_part[XE_NEUTRAL][n2]);
        goto end_select_n2;
      }
      else
        continue;
    }
    else
      nrerror("Failed selection in collision, n2\n");
```

The collision cross section used in (3.5-1) depends on the collision partner selected. If there is more than one possible collision type (a Xe-Xe+ pair can undergo either elastic or CEX collisions), the type of cross section is also chosen randomly and then calculated based on the particle's relative velocities.

As a final note, the selection rejection scheme is known to fail under some circumstances. One case is when a single large value of Q makes Q_{max} much larger than Q_{mean}. Since (3.5-1) is based on the ratio of Q to Q_{max}, a very large Q_{max} will cause the simulation to spend most of its time rejecting collision pairs. Some cross sections, like the Coulomb cross section, approach infinity as the relative velocity between the two collision partners approaches zero. These collisions can not be simulated using the selection-rejection method because the large cross section results in $Q_{max} >> Q_{mean}$. The selection-rejection scheme also fails if there are too few particles in the cell. The simulation can, for instance, take two particles and repeatedly collide them together in a single timestep. This results in little effect after the first few collisions. Empirically, it has been observed that about 20 particles are required in each cell to simulate the dynamics of neutral collisions properly. The simulation includes a hard limit which ensures that no collision are carried out if fewer than 4 neutrals and 4 ions/double ions are present in a given cell. This criterion is discussed more in the next section.

Time Counter

As was discussed in Chapter 1, the time counter for a single species undergoing collisions with itself is given by

$$\Delta t_c = \frac{2}{Vn^2\sigma c_r} \tag{3.5-3}$$

The number density is given by n = NW/V where N is the number of macro particles in the cell, V is the cell volume, and W is the macroparticle weighting factor. (3.5-3) is the time increment for each real collision. Each simulated collision represents W real collisions, so the final expression for the time increment for each simulated collision is

$$\Delta t = \frac{2V}{N^2W\sigma c_r} \tag{3.5-4}$$

Equation (3.5-4) is valid for species undergoing collisions with themselves, regardless of the number of species in the simulation. One approach to simulating multiple collision between multiple species is to maintain a separate time counter for each type of collision. The general form of (3.5-4) is simply

$$\Delta t_{cij} = \frac{\left(1+\delta_{ij}\right)}{n_i n_j V\sigma_{ij}c_r} \tag{3.5-5}$$

Where δ is the Kronecker delta and i and j are indices representing each type of particle in the simulation. Since a copy of each time counter must be stored with each grid cell, the memory required to store time counters rapidly comes to exceed the memory used to store the grid itself. *Elgin* showed that in a multi-species mixture in which each species can interact with itself or with other species, a single local time counter can be used for each cell. The time increment for each real collision is then given by

$$\Delta t_g = \frac{2}{Vn^2\sigma_{ij}c_r} \tag{3.5-6}$$

Where n is the total number density and is given by

$$n = n_1 + n_2 + \ldots + n_i$$

Elgin's time counter is valid when each species will undergo one and only one type of collision with each of the other species in the simulation. In this case, however, some species undergo no collision interactions (Xe^+-Xe^{++} for instance) while other species can undergo multiple interactions (Xe-Xe^+ CEX and Xe-Xe^+ elastic collisions). We therefore derive a different multi-species time counter using the following procedure.

Equation (3.5-5) can be used to simulate multiple collisions by maintaining a separate counter for each collision pair. But as long as collisions occur in the correct proportion to each other, it is sufficient to monitor only one collision counter. The difficulty is in selecting a counter to monitor. If once chose to monitor collisions of type 2-2, for example, the system would fail in cells where no particles of type 2 were present. This problem is avoided by monitoring an average of all of the counters. Given that one can monitor any one time counter, it follows that one can also monitor any weighted average of the time counters. This results in a single multi-species time counter with the following form

$$\Delta t = \frac{\sum_{i=0}^{s}\sum_{j=1}^{i} D_{ij} t_{cij}}{\sum_{i=1}^{s}\sum_{j=1}^{i} D_{ij}} \qquad (3.5\text{-}7)$$

This time counter is valid for any D. *Elgin* uses

$$D_{ij} = \frac{n_i n_j}{\left(1+\delta_{ij}\right)} \qquad (3.5\text{-}8)$$

which gives (3.5-5) above when all species can collide with all other species in one and only one way. In this case, we are considering five different types of collision occurring between three species. Combining (3.5-7) and (3.5-8) in this case gives

$$\Delta t = \frac{n_o n_1 t_{01CEX} + n_o n_1 t_{01E} + n_0 n_2 t_{02CEX} + n_o n_2 t_{02E} + {}^{n_o t_{00}}\!/_{2}}{2n_o n_1 + 2n_o n_2 + {}^{n_1^2}\!/_{2}} \qquad (3.5\text{-}9)$$

Where n_0 is the neutral density, n_1 is the Xe single ion density, n_2 is the Xenon double ion density, and t is the collision counter for that particular combination as given by (3.5-5). So, for instance, Δt_{01CEX} is given by

$$\Delta t_{01CEX} = \frac{1}{V n_1 n_1 \sigma_{12CEX} c_r}$$

Substituting and simplifying gives the following expression for the multi-species time counter

$$\Delta t_g = \frac{1}{\left(2n_0 n_1 + 2n_0 n_2 + {}^{n_1^2}\!/_{2}\right) V \sigma c_r}$$

where V is the volume of the cell. Finally, we must account for variable macroparticle weighting factors. In each case, the local number density is given by an expression of form

$$n_1 = \frac{N_1 W_1}{V}$$

Substituting gives this form for the collision time counter

$$\Delta t = \frac{V}{N_0 W_o \left(2N_1 W_1 + 2N_2 W_2 + \frac{N_0 W_0}{2} \right) \sigma_{ij} c_r}$$

This is the time increment for each real collision. The time increment for each simulated collision is simply

$$\Delta t = \frac{W_L V}{N_0 W_o \left(2N_1 W_1 + 2N_2 W_2 + \frac{N_0 W_0}{2} \right) \sigma_{ij} c_r} \qquad (3.5\text{-}10)$$

This is the form of the time counter implemented in the PIC-DSMC simulation.

Expression (3.5-10) can be expanded to include additional species and additional collisions. There are, however, practical limits on the application of (3.5-10) in a simulation. The magnitude of expression (3.5-10) depends linearly on the lowest weight of the two particles undergoing collision. If $N_1 >> N_2$, 1-1 collisions will have timesteps much greater than 1-2 collisions (assuming that the two collision cross sections are roughly the same order of magnitude). If a timestep is chosen so that only one collision of type 1-2 occurs in each iteration, when a collision of type 1-1 occurs (which will happen, despite the low selection-rejection probability), the local time will be set to a value much larger than the global time step. This effectively removes the cell from consideration for many timesteps. If the timestep is set so a collision of type 1-1 occurs in each timestep, the simulation will also carry out many collisions of type 1-2 in each iteration. This is very computationally inefficient. Expression (3.5-10) is therefore practical only when the macroparticle weights used for different species are the same order. The importance of this issue has been recognized in the past and some authors have proposed "non-time counter" schemes which may avoid some of the problems [*Baganoff et al.* 1990]. These techniques represent the state of the art in DSMC and have not been incorporated into this work.

Very large timesteps will also occur when very few macroparticles are present in a cell. When simulating vacuum conditions, much of the simulation will initially contain few particles, resulting in very large local timesteps. As the simulation continues, however, particles entering these cells will be unable to collide with each other because the local time is much greater than the simulation's global time counter. To avoid this problem, a numerical floor is placed on the number of particles in each cell. If fewer than four neutral or four charged (ion + double ion) macroparticles are present in a cell, the local time is set equal to the global time and the simulation proceeds to the next cell. This floor is consistent with the behavior of physical systems. When very few particles are present, no collisions occur within the cell. But if the number of particles in the cell increases, collisions immediately begin to occur. In theory, this technique will cause problems if the number of particles in the cell continually oscillates around the limiting value. In practice, no problems of this type have been observed in the plume simulation.

The need to keep particles on approximately the same timescale limited our ability to model Xe-Xe collisions occurring in experimental vacuum tanks. The neutral density in a vacuum tank is much higher than the ion density in the plume. As a result, $W_0 >> W_1$ and W_2 when simulating these laboratory cases. This causes a timescale conflict between neutral-neutral and CEX collisions. So, for computational reasons, we omit Xe-Xe from simulations of laboratory cases. The physical consequences on the plume are negligible. Xe-Xe collisions serve largely to thermalize the background neutrals, and these are already maxwellian by assumption. Xe-Xe collisions were included in simulations of thrusters operating in vacuum. In those cases, $W_0 \sim W_1 \sim W_2$, so no computational problems were encountered. As stated earlier, these collisions were included for completeness and to allow the simulation of thrusters with higher neutral flows. The results shown in Table 2.2 demonstrate that these collisions play little role in the development of Hall Thruster plumes, and simulations conducted with Xe-Xe collisions produce virtually identical results to those conducted without Xe-Xe collisions.

Coulomb collisions between charged particles are also omitted from the present work for computational reasons. Coulomb collisions are extremely difficult to simulate using particle methods because the Coulomb cross section is relatively large and the collisions dynamics are dominated by small angle collisions. Each individual collision results in little momentum transfer, but significant momentum transfer does occur over the course of many collisions. Simulating Coulomb collisions directly would involve simulating many small collisions to determine the net momentum transferred to each macroparticle. However, it is much more efficient to simulate a few collisions which drastically change the trajectory of the collision partners than it is to simulate many collisions which slightly change the trajectory of the collision partners, even if the net effect on the plasma is the same. The direct modeling of Coulomb collisions is therefore highly inefficient and is probably beyond the capability of current (1996) computer workstations. This should have little effect on the structure of the plume itself. In Table 2.2, it was shown that the ion-ion mean free path is quite long in the plumes of Hall thrusters. It should be noted, however, that increasing the plasma density by a single order of magnitude might result in an ion-ion mean free path short enough to have some effect on the plume region. The effects of Coulomb collisions should be considered carefully before modeling plumes with higher plasma densities.

Although Coulomb collisions are not considered in the present work, it is worth discussing some of the approaches which might be used to simulate them in future work. Since it is easier to simulate a few large angle collisions than it is a large number of small angle collisions, an interesting approach would be to determine the average effect of the small angle collisions and treat them as a few large angle collisions instead. This is similar to the methods used to simulate collisions in conventional DSMC. Since the initial position of the particles is unknown, their final state is chosen randomly based on a distribution function that reflects the average behavior of the particles in the gas. With hard sphere collisions, for instance, the scattering angle is isotropic, so the particles' exit angles are chosen from an isotropic distribution. The problem is therefore to determine a distribution of exit angles which are equivalent to the distribution created by many small angle collisions. This would require the use of some type of statistical averaging, perhaps similar to that used to solve the Vlasov and Boltzmann equations. These formulations model the evolution of the distribution function in phase space. Coulomb collisions appear as a collision operator which is constructed by averaging the effects of many small angle collisions into a single non-linear term. This operator is known as the Fokker-Planck collision operator. The problem is that the Fokker-Planck operation is constructed by averaging over the distribution function. Particle simulations, on the other hand, work at a microscopic level, and the local distribution function plays no direct role in any individual collision. Simulating Coulomb collisions may involve constructing some sort of exit angle distribution based on the local distribution function. In principle, this function can be constructed by looking at the particles in the cell. In practice, this function is very noisy and difficult to resolve without the use of time averaging. Further work is clearly needed in this area to resolve the various theoretical and computational issues surrounding this approach. At the present time, it is not clear that small angle collisions can be effectively simulated using particle methods.

3.5.2 Collision Cross Sections and Dynamics

When a collision takes place between two simulated particles, the dynamic of the collision are very similar to those of real particles. The collisions themselves are not entirely deterministic, however, because the initial position of the particles relative to each other is unknown. This section discusses the collision dynamics used in the axisymmetric plume model and the collision cross sections used in the selection-rejections process (as discussed in section 3.5.1). A total of five different types of collisions are modeled in the simulation, and each of these collisions is discussed in turn in this section. Several of the relevant collision cross sections were derived in section 2.2.2, so details of their origin are not described in this section.

Charge Exchange Collisions

The collision process which most affects the structure of the plume is charge exchange (CEX). Ion-neutral charge exchange occurs when a neutral and an ion exchange an electron through a resonant collision process. Neutral-double ion charge exchange and single ion-double ion charge exchange can also occur in some circumstances. The collision cross sections used in DSMC portion of the simulation are physical cross sections based on theory and on laboratory measurements. The Xenon neutral-Xenon single ion CEX collision cross section has been measured experimentally (see Figure 2.8) and has been calculated by *Rapp and Francis* [1962]. They give the collision cross section for resonant charge exchange between Xenon neutral and Xenon ions as [*Samanta Roy* 1995]

$$\sigma_{\text{CEX}} = \left(k_1 \ln c_r + k_2\right)^2 \cdot 10^{-20}\,\text{m}^2 \qquad (3.5\text{-}11)$$

Where $k_1 = -0.8821$, $k_2 = 15.1262$, and c_r is in m/s. This form of the cross section is used in the PIC-DSMC plume model.

Two other types of CEX were considered for the PIC-DSMC plume model. The Xe-Xe^{++} (two electron transfer) and Xe^{+}-Xe^{++} CEX cross sections have both been measured experimentally and are shown in Figure 3.13 [*Hasted* 1964]. Xe-Xe^{++} are included in the plume model. The collision cross section used in the simulation is a logarithmic fit to data shown in Figure 3.13. The logarithmic fit results in the following expression.

$$\sigma_{\text{CEX}} = \left(3.4069\times10^{-9} - 2.7038\times10^{-10}\ln c_r\right)\,\text{m}^2$$

Figure 3.13 also shows the cross section for CEX between Xe^{+} and Xe^{++}. This cross section is almost an order of magnitude smaller than the Xe-Xe^{++} CEX cross section, so we choose to neglect Xe^{+}-Xe^{++} CEX in the simulation.

The dynamics of Xe-Xe^{+} and Xe-Xe^{++} CEX are treated identically in the plume simulation. A physical CEX collision takes place when one (or two) electrons jumps from an ion to a neutral, thus changing the neutral into an ion and the ion into a neutral. In a physical collision, the electron jumps from one particle to another. In a DSMC collision, however, the position of the particles themselves is irrelevant to the collision process. It is therefore just as valid to have the two partners trade velocities as it is to have them transfer an electron. From a computational point of view, this is also more efficient because it requires the transfer of less information. We therefore simulate the dynamics of CEX collisions by simply switching the velocity vectors of the ion and neutral collision partners (subject to the second selection-rejection criterion discussed in section 3.5.1).

Ion-Neutral Elastic Collisions

Xe-Xe^{+} and Xe-Xe^{++} elastic collisions are two additional types of collisions included in the plume model. The cross section for Xe-Xe^{+} collisions has been calculated by *Banks* and given in section 2.2.3 as

$$\sigma_{\text{in}}(c_r) = 2.21\pi\left(\frac{\psi e^2}{m_{12}c_r^{\,2}}\right)^{1/2}$$

This cross section is of the "Maxwellian Collision" type for which collision frequency is velocity independent. Filling in the polarizability of Xenon and evaluating this expression in SI units gives

$$\sigma_E = \left(8.28072\times10^{-16}/c_r\right)\,\text{m}^2$$

According to theory, the ion mobility is independent of charge state [*Dalgarno* 1958]. The collision cross section therefore scales directly with the ion charge, so the Xe-Xe++ elastic cross section is equal to twice the Xe-Xe+ cross section.

The details of the collision dynamics of Xe-Xe+ collisions are complex. We have chosen to simplify the collision dynamics and treat them as though the particles were undergoing hard-sphere collisions. Because the collision cross section varies with velocity, this method can be regarded as a variation of the Variable Hard Sphere (VHS) collision method described in *Bird* [1994, Ch. 2]. The result of a collision between two particles can be determined from their initial velocity and initial position based on the conservation of energy, conservation of momentum, and detailed information about the nature of the impact. Consider two particles of mass m_1 and m_2 which are traveling at velocities $\mathbf{c}_1$ and $\mathbf{c}_2$ and undergo a collision. The collision will results in post-collision velocities of $\mathbf{c}_1^*$ and $\mathbf{c}_2^*$ which are related to $\mathbf{c}_1$ and $\mathbf{c}_2$ by conservation of momentum and conservation on energy, i.e.

$$m_1\mathbf{c}_1 + m_2\mathbf{c}_2 = m_1\mathbf{c}_1^* + m_2\mathbf{c}_2^* = (m_1 + m_2)\mathbf{c}_m$$
$$m_1\mathbf{c}_1{}^2 + m_2\mathbf{c}_2{}^2 = m_1\mathbf{c}_1^{*2} + m_2\mathbf{c}_2^{*2}$$

Where $\mathbf{c}_m$ is the velocity of the center of mass of the two collision partners. $\mathbf{c}_m$ is a conserved quantity given by

$$\mathbf{c}_m = \frac{m_1\mathbf{c}_1 + m_2\mathbf{c}_2}{m_1 + m_2} \tag{3.5-12}$$

The pre-collision velocities can be expressed in terms of $\mathbf{c}_m$ using the following expressions

$$\mathbf{c}_1 = \mathbf{c}_m + \frac{m_2}{m_1 + m_2}\mathbf{c}_r \tag{3.5-13}$$
$$\mathbf{c}_2 = \mathbf{c}_m - \frac{m_1}{m_1 + m_2}\mathbf{c}_r$$

Where $\mathbf{c}_r$ is the relative velocity of the two particles and is given by

$$\mathbf{c}_r = \mathbf{c}_1 - \mathbf{c}_2 \tag{3.5-14}$$

Figure 3.14 shows the trajectory of two particles undergoing a collision in the center of mass coordinate system. As shown in (3.5-13), the collision is planar in a center of mass frame of reference. Since $\mathbf{c}_m$ is fixed, the trick is to figure out $\mathbf{c}_r^*$. Once that is known, the post-collision velocity is given by

$$\mathbf{c}_1^* = \mathbf{c}_m + \frac{m_2}{m_1 + m_2}\mathbf{c}_r^*$$
$$\mathbf{c}_2^* = \mathbf{c}_m - \frac{m_1}{m_1 + m_2}\mathbf{c}_r^* \tag{3.5-15}$$

From energy conservation, the magnitude of the relative velocity between the two particles is unchanged by the collision. i.e.

$$c_r^* = c_r$$

Therefore, the only unknown quantity is the change in the particle's direction, or the angle χ as shown in Figure 3.14. χ can be determined directly given the particles' distance of closest approach (b) and a knowledge of the interaction potential [*Bird* 1994, Ch. 2]. The particles' initial position is needed to determine the distance of closest approach of the undisturbed trajectories in center of mass coordinates. This quantity is commonly referred to as the collision parameter. In a DSMC collision, the initial position of the particles is considered unknown because the physical location of the particles in the simulation has no bearing on the physics of the collision. The two trajectories may not even intersect each other in the computational space. As a result, the collision parameter is also unknown and must be chosen randomly based on a model of the collision dynamics. Since b and χ are directly related to each other , this is equivalent to choosing χ based on the collision dynamics. For hard sphere collisions the scattering angle is isotropic, which is to say that all directions are equally likely for $\mathbf{c}_r{}^*$. The procedure for the variable hard sphere model is therefore to randomly pick a direction for $\mathbf{c}_r$ using a procedure described by *Bird* [1976, pg. 131].

$$\begin{gathered}\cos\theta = 1 - 2\beta \\ \phi = 2\pi\beta \\ \sin\theta = \sqrt{1-\cos^2\theta}\end{gathered} \qquad (3.5\text{-}16)$$

$$\mathbf{c}_r^* = \mathbf{c}_r \begin{bmatrix} \cos\theta \\ \sin\theta\cos\phi \\ \sin\theta\sin\phi \end{bmatrix}$$

Where β represents a randomly chosen number between [0:1]. The collision sequence proceeds as follows. After the collision partners have been chosen, their relative and center of mass velocities are calculated using equations (3.5-12) and (3.5-14). The new relative velocity is calculated based on two randomly chosen numbers, as shown in sequence (3.5-16). The post-collision velocities are then calculated from equation (3.5-15), and the particle vectors are modified according to the Monte-Carlo sequence described in section 3.5.1.

Neutral-Neutral Elastic Collisions

Finally, Xe-Xe elastic collisions are also included in the present plume simulation in cases where the background is a vacuum. The inclusion of neutrals allows one to model hypothetical thrusters with high neutral flow rates. Neutral-neutral collisions have been studied extensively in the literature, and a variety of different techniques have been developed to simulate them efficiently. These techniques include the well known hard sphere, variable hard sphere, soft sphere, and variable soft sphere models [*Bird*, 1994, Ch. 2]. The variable soft sphere (VSS) model contains the most degrees of freedom and therefore generally provides the best fits to laboratory data. We use a standard VSS model for the Xe-Xe neutral cross section and collision dynamics. In the VSS model, the diameter of the collision varies with the particles approach velocity according to a simple inverse power law of the form

$$d = d_{ref}\left(c_{r,ref}/c_r\right)^{\upsilon}$$

where d_{ref} and c_{ref} are a reference diameter and a reference approach speed and ν is an experimentally determined constant. It has been shown that the ν is given by a gas's experimentally measured variation in the coefficient of viscosity with temperature [*Bird* 1994, Ch. 2]. The reference values and constants for a variety of gases are presented by *Koura* [1992]. Based on measurements of transport properties made at temperatures from 300 K - 15000 K, *Koura* obtains the following cross section for Xenon gas (after conversion to SI units)

$$\sigma_E = \left(2.117\times10^{-18} c_r^{-0.24}\right)\,\mathrm{m}^2 \qquad (3.5\text{-}17)$$

The model for the collision dynamics is similar to the hard sphere model used for Xe-Xe+ and Xe-Xe++ collisions. The only difference is that the scattering angle is not isotropic, but is of the form

$$\chi = 2\cos^{-1}\left[2(b/d)^{1/\alpha}\right]$$

where b is the collision parameter, d is the diameter of the molecule based on equation (3.5-17), and α is an experimentally determined constant based on the temperature dependence of the coefficients of viscosity and diffusion. Values for α are also presented in *Koura* [1992]. When implemented computationally, the scattering angle takes the following form

$$\alpha = 1.107 c_r^{0.01944}$$

$$\chi = \cos^{-1}\left(2\beta^{1/\alpha} - 1\right) \qquad (3.5\text{-}18)$$

The new velocity vectors are determined using the equations for conservation of momentum and energy, as described above in the section on Ion-Neutral Elastic collisions.

3.6 Surface Interaction Models

The axisymmetric version of the simulation was intended to verify the basic plume model and to simulate very simple thrust geometries. As a result, the only surfaces generally present in the simulation are the surfaces of the thruster itself. Relatively little is known about these surfaces and their potential with respect to the plume plasma. In many cases, the exact composition of these surfaces is also unreported in the literature. Because of these uncertainties, we chose not to implement a full surface interaction model in the axisymmetric simulation. Instead, we constructed two separate surface models, one which provides boundary conditions along the surfaces of the thruster, and another which calculates erosion rates along a virtual surface placed at a constant distance from the thruster exit. This section describes the surface models used in the axisymmetric simulation. Section 3.6.1 describes the boundary conditions imposed on surfaces of the thruster and section 3.6.2 describes the sputter erosion model. A fully integrated surface-interaction package was also developed for the three dimensional plume model. This surface model is described in detail in Chapter 5.

3.6.1 Interior Boundary Conditions

When a non-conducting surface is exposed to a plasma, the potential of the surface is determined by the flux of ions and electrons to the surface. If the surface were sitting at the plasma potential, the flux of electrons and ions to the surface would be given by equation (3.3-3), i.e.

$$\Gamma = \frac{n\bar{c}}{4} = \frac{n}{4}\sqrt{\frac{8kT}{\pi m}}$$

In a Xenon plasma, the electron mass is six orders of magnitude less than the ion mass, so the flux of electrons to the surface is much higher than the flux of ions. When charged particles strike a solid surface, they are generally neutralized. Electrons striking the surface are absorbed and enter the conduction band. Ions are neutralized just above the surface by electrons emitted from the surface. As the surface collects electrons, it acquires a negative potential and begins to repel electrons away from the surface. This results in the formation of a non-neutral sheath with a width of the same order as the Debye length (as shown in Figure 3.15). Eventually, the electron flux just balances the ion flux, resulting in a steady state. The resulting surface potential has been calculated for a variety of different geometries. For a planar surface, it is given by [*Chen* Ch. 8]

$$\Phi_f = -\frac{kT_e}{e}\ln\left(\frac{4\Gamma_i}{n_o\bar{c}_e}\right) \tag{3.6-1}$$

where $\bar{c}_e$ is the electron thermal speed as given in (3.3-3). Because the sheath is a non quasi-neutral region, it can not be directly simulated using a quasi-neutral PIC formulation. Instead, analytic boundary conditions must be imposed at the sheath boundary. The pre-sheath region can be simulated using a quasi-neutral formulation. In this region, the ions are accelerated to the Bohm velocity (roughly equivalent to the ion sonic velocity) or higher before they enter the sheath. Because the potential of the sheath is less than that of the pre-sheath, all ions entering the sheath end up being neutralized at the wall. The sheath boundary therefore resembles a sink which ions enter but from which they never emerge again. Because the ions are traveling at sonic velocities, no information about the sheath can travel into the bulk plasma. In addition, Debye shielding ensures that the bulk of the plasma does not see the negative wall potential. The non quasi-neutral region is limited to a thin region near the wall with a width of the same order as the Debye length.

Based on these observations, a simple set of boundary conditions can be constructed based on the following assumptions

- The potential of the surface is less than the plasma potential.
- The width of the sheath is much less than the length scale of the solid bodies simulated.

The first assumption is generally valid for dielectric surfaces. The second assumption is necessary to justify use of the planar sheath model shown in Figure 3.15. This model is valid for any surface in which features of interest are generally larger than the sheath width. The width of a planar sheath is generally of same order as the Debye length and is given by the Child-Langmuir relationship [Chen, pg. 294]

$$j = \frac{4}{9}\left(\frac{2e}{m_i}\right)^{1/2} \frac{\varepsilon_o|\phi_w|^{3/2}}{d^2}$$

Where j is the ion current to the wall and d is the width of the sheath region. If ions approaching the wall are traveling at the Bohm velocity, the current to the wall is given by

$$j = en_e\sqrt{\frac{kT_e}{m_i}}$$

Setting these two expressions equal to each other and rearranging gives the following relationship for the width of the sheath region

$$\frac{d}{\lambda_d} = \frac{2}{3}2^{1/4}\left(\frac{e\phi}{kT_e}\right)^{3/4}$$

So one can conclude the following

$$w \sim \phi^{3/4}$$

Where w is the width of the sheath region. Table 2.2 shows that the Debye length in the plume is generally of order 1 mm, so meter scale objects can be accurately represented using planar sheath theory at all but very high potentials. In the planar sheath model, ions enter the sheath at sonic velocities and never emerge. As a result, it is not necessary to directly simulate the sheath region. Instead, boundary conditions can be imposed at the sheath/pre-sheath boundary. Ions striking this boundary are removed from the simulation, thus creating a sink region. The presence of the sink causes the pre-sheath potential to adjust itself to fulfill the Bohm sheath criterion. The potential of the surface is not important as long as the surface potential is less than the pre-sheath potential. The pre-sheath potential is typically of the same order as the electron temperature, which in this case is 2 Volts. It should be noted that fine features such as solar cell interconnectors may not be correctly simulated by a planar sheath model. These features may have a significant impact on charge collection and could cause undesirable effects such as solar cell arcing [*Cho* 1991].

In practice, the planar sheath boundary condition is imposed by treating the solid surfaces as absorbing boundaries when struck by ions. The grid cell boundary represents the sheath/pre-sheath boundary and acts as a charge sink. The exact potential of the surfaces is unimportant and is not calculated in the axisymmetric plume model. A more sophisticated model which includes a crude model for the effects of positive boundary potentials is used in the three dimensional plume model. This model is discussed in detail in Chapter 5.

When neutral molecules strike solid surfaces, they return to the gas after interacting with the upper levels of molecules in the surface. This results in some level of accommodation with the surface which can be simulated using different accommodation coefficients [*Bird* 1994, pg. 118]. In most Hall thruster experiments, both the temperature of the surface and the accommodation coefficients are unknown. We therefore treat solid surfaces as reflective surfaces when struck by neutral particles. This is known as specular reflection, and is equivalent to assuming an accommodation coefficient of zero. When the temperature of the surface were known, a better limit to use would be full accommodation with diffuse scattering. This would better reflect the transfer of energy from the particle to the molecules on the solid surface. In this case, we have chosen specular reflection simply because the temperature of the surfaces are not known.

3.6.2 Surface Erosion Model

Previous work has shown that eroded metal from the acceleration grids of ion thrusters can lead to substantial deposition on solar cell cover glasses [*Samanta Roy* 1995]. The anode of an SPT thruster also undergoes substantial erosion and releases a ceramic Boron Nitride (BN) and/or its components (Boron and Nitrogen) compound into the plume region. These materials are fairly inert, and experiments show no evidence that it deposits onto surfaces [*Randolph* 1994]. Since the propellant is also a noble gas and is unlikely to deposit onto surfaces, plume induced deposition is neglected in our surface interaction model. That assumption is probably not be valid for the TAL thruster, which has a metal anode. This assumption may also be invalid if the anode erodes to a point where the metal poles of the magnets are exposed to the plume. Under those conditions, experiments suggest that a small amount of deposition may occur in some parts of the plume [*Pencil* 1996]. Both neutrals and ions can contribute to erosion due to sputtering. Calculating sputtering rates requires a knowledge of the wall potential, so these rates are not calculated on most surfaces in the axisymmetric plume model. Instead, erosion rates are calculated on a virtual surface forming an arc in front of the thruster. This section discusses the model used to estimate surface erosion rates and the assumptions behind this model.

Sputtering erosion rates are calculated by recording the flux and energy of macroparticles crossing an arc at a fixed distance from the anode exit. The surface is referred to as a virtual surface because the ions are not neutralized by the surface, but continue on as though the surface were not present. This causes the simulation to slightly underestimate the acceleration of ions in the pre-sheath and ignores the effects of returns neutrals and of the sputtered atoms themselves. The presheath potential is on the order of the electron temperature, which in this case is 2 volts. This drop is small with respect to the velocity of the beam ions, which are moving at velocities in excess of 10000 m/s. Neglecting this drop has little effect on the final result. After the run is completed, the flux and energy information are post-processed to determine the magnitude of the sheath drop which would be present on a surface sitting at the location of the arc. The magnitude of this drop is given by equation (3.6-1). The electron thermal speed and temperature are determined based on an isothermal Maxwellian electron distribution with a temperature of 2 eV.

Once the sheath drop has been calculated, the energy that each particle would obtain from the sheath is then added to the recorded macroparticle energy to give the impact energy of ions on the surface. Once the impact energy is known, the amount of material removed can be determined the macroparticle weight by an experimentally measured energy dependent sputtering coefficient. This coefficient is material dependent and must be determined by experiment. Since Xenon is the dominant species in the plume region, only Xenon induced sputtering is considered in the model. One material of interest to designers is silver, which is used to make solar cell interconnectors. The sputtering coefficient is defined as the average number of atoms sputtered by each ion that strikes a surface. The sputtering coefficient for Xenon ions striking silver at angles normal to the surface was measured experimentally by *Rosenberg* [1962] and is shown in Figure 3.16. Also shown is a linear fit to the data based on the following expression

$$S = 7.334 \times 10^{-3} E - 0.29511 \tag{3.6-2}$$

The agreement between the fit and the data is very good, and equation (3.6-2) is used to represent the sputtering coefficient in both the axisymmetric and three dimensional surface models. Another material of interest is the glass used to cover solar cells. It is relatively difficult to obtain the sputtering coefficient for a given solar cell coverglass. Many companies consider this information proprietary, and solar cell coverglasses are often covered by a thin anti-reflective coating whose sputtering coefficient is also unknown. We therefore use a linear fit to the sputtering coefficient for Argon on quartz glass to represent a generic glass surface. It is given by [*Randolph* 1994]

$$S = 7.105 \times 10^{-4} E - 0.01815 \tag{3.6-3}$$

Quartz is tougher than most glasses, so this expression underestimates the actual sputtering coefficient (see Chapter 4). This fit is also used in the three dimensional plume model. Only two surfaces are included in the present simulation. In principle, however, any surface of known sputtering coefficient can be simulated by using correct values for the sputtering coefficient. At the present time, no effort is made to calculate the effects of ion impact angle on the sputtering coefficient. In reality, the sputtering coefficient depends strongly on impact angle. Figure 3.17 gives a generalized picture of the dependence of the sputtering coefficient on impact angle. Because the peak sputtering coefficient typically occurs at non-normal impact angles, (3.6-2) and (3.6-3) may underestimate the actual erosion of surfaces due to sputtering. Unfortunately, very few experimental measurements have been made of Xenon sputtering coefficients and their dependence on impact angle. Without further experimental work, it is not possible to include such effects in our computational model.

In summary, a surface interaction model has been developed for the axisymmetric plume simulation. This model calculates sputtering rates on two surfaces, silver and quartz glass, but can be extend to calculate sputtering on any surface for which the energy dependent sputtering coefficient is known. The model records the flux of ions crossing a virtual surface and then calculates the potential drop across the sheath based on that flux. It then calculates the sputtering rate based on an energy dependent sputtering coefficient.

3.7 Computational Notes (Memory Management)

For decades, the preferred computer programming language for engineers has been the FORTRAN 77 language, which is easy to learn and has, in the past, provided superior performance. The simulation described in this document was written in C, the language that forms the basis for UNIX systems. C provides programmers with more options than FORTRAN 77. One of the most useful options is the ability to dynamically allocate and deallocate memory as it is needed. This allows programmers to use advanced data structures and memory storage methods. This section discusses the linked list memory management scheme used in the PIC-DSMC simulation and compares it to conventional storage schemes. Our experience suggests that linked lists are not a superior storage scheme because they add a high level of complexity to the simulation for little performance gain.

Conventional PIC simulations store particles in large arrays and use "garbage collection" routines to repack the arrays as particles are removed from the simulation. This technique is shown schematically in Figure 3.18. At the beginning of the simulation, a large block of memory is set aside for particle storage. As particles are created, the array is slowly filled. As the simulation runs, particles are both created as they enter the domain and destroyed as they exit the domain. The destroyed particles create gaps in the memory array. These gaps are marked as empty and are skipped in particle move and weight routines. As the number of gaps increase, the memory usage becomes more inefficient and more particles are skipped in each timestep. After a number of iterations, a "garbage collection" routine is used to repack the array, removing the empty particles and replacing them with active particles from further down the list. This process is repeated until the simulation is complete. The maximum number of particles in the simulation is fixed by the size of the initial memory block. If the number of particles in the simulation exceeds the initial memory allocation, the simulation fails. The main advantage of this memory storage method is that it is relatively simple and can be implemented in languages like FORTRAN 77 that do not allow dynamic memory allocation. The main disadvantages are that the amount of memory required is fixed by the programmer (so the program must occupy large areas of memory even at the beginning of the simulation when relatively few particles may be present) and that there is substantial overhead associated with the garbage collection routine.

Our simulations use a linked list data structure in which particles are stored in separate areas of memory and are linked together dynamically. This structure is shown schematically in Figure 3.19. The basic scheme involves a data structure for each particle that looks like this

```
typedef struct _particledata{
  float x;                      /* X position */
  float y;                      /* Y position */
  float vx;                     /* X velocity */
  float vy;                     /* Y velocity */
  float vz;                     /* Z velocity */
  char species;                 /* Species Identifier */
  char label;                   /* Particle label (for visualization only) */
  char index;                   /* Mesh index for finest species */
  int i;                        /* Position within finestmesh */
  int j;                                             /* Pointer to next element */
  struct _particledata *next;
  struct _particledata *cell_next;  /* Pointer to next particle in same cell */
  struct _particledata *cell_prev; /* Pointer to prev. particle in same cell */
                                                     /* Particle state vector */
 } particledata;
```

Each particle is stored as a data structure that not only contains its state vector and identifier, but also includes two pointers that point to two other particles. At the beginning of the simulation, the list consists of a single particle. When a new particle is created, memory is allocated for the new particle and a new state vector and identifier are created. Then, the particle is linked to the first particle in the list, and the first particle in the list is linked back to the new particle (as shown in Figure 3.19). The new particle then becomes the first particle in the list. When a particle is destroyed, the particle is removed from the list and the particles in front of it and behind it are linked to each other, thus restoring the chain of linked particles (see Figure 3.19). The memory representing the destroyed particle can then be freed and removed from the simulation. Linked lists are commonly used to store long lists of data from which items are continuously removed and added. This makes a linked list a natural structure for storing particles in a PIC or DSMC simulation. One problem with linked lists is that it is difficult to access an element in the middle of the chain without accessing those at the beginning or end of the chain first. This is not generally a problem for PIC simulations because particles can be accessed serially one after another in the move and weighting steps. In the PIC-DSMC model, the boundary conditions are also evaluated during the move phase, so there is never a need to examine particles at the center of the chain.

A linked list structure has several advantages for particle-based simulations. First, memory is dynamically allocated and freed as the simulation runs, so there is no upper limit on the number of particles allowed (except for physical RAM and disk space) and the simulation never occupies more memory than it actually requires at any given point in time. A linked list is also a very natural structure for the particles since they are generally accessed in order and since particles are continuously created and destroyed as the simulation runs. There are also some disadvantages associated with linked lists. Each particle in a linked list actually occupies slightly more memory that particles in a block array because two pointers are required in addition to the state vector and identifier information. A linked list is also more complex than a garbage collection scheme, which makes it more difficult to implement and debug. The main disadvantage for scientific simulation has to do with workstation performance issues. At the present time, the speed of central processing units in engineering workstations exceeds the speed of the memory chips and the buses used to access them. As a result, memory access issues often limit the computation speed of modern workstations. To work around the memory access problems, most modern computers include some very fast "cache" memory which can be accessed quickly by the CPU. When the CPU requests a value, the computer actually downloads a large section of memory into cache. Values which are stored near the initial value can then be accessed very quickly, without going through the main bus into relatively slow general storage memory. In a conventional "garbage collection" scheme, particles are evaluated in the order in which they are stored, the cache is used efficiently when moving or weighting the particles. One disadvantage of a linked list is that because particles are allocated and freed dynamically, particles which are located next to each other in the linked list are not necessarily located next to each other in memory. As a result, there can be considerable overhead associated with trading memory in and out of cache in order to process the next member in the list. This has been observed to have a considerable impact on the performance of the simulation.

In order to mitigate the effect of "cache hits," the memory management scheme used in our model is a modified linked list which initially allocates memory in blocks large enough to hold 200 particles each. When particles are added to the simulation, the new memory is taken from the next free space in the block, so particles which are adjacent to each other in the list are also adjacent to each other in memory. This scheme works well initially, before particles have been removed from the simulation, but when particles are removed from the list, the "freed" memory is reused by the next new particle. Over time, the memory used in the list is scrambled by the creation and destruction of particles. Although all of the memory in the list is stored in continuous memory, particles next to each other in the list are no longer stored next to each other in memory. This memory management scheme is therefore only a partial solution to the cache hit problem. It has been observed to cut execution times by about 10%.

The routines used to implement the modified linked list are relatively complex. In addition, because the use of non-adjacent memory incurs a performance penalty, a garbage collection scheme may actually provide better performance for particle computations. The linked list is a natural and elegant data structure for this problem. However, the complexity of the scheme and the lack of any noticeable improvement in the performance of the simulation lead us to recommend that future implementations continue to use a conventional garbage collection scheme.

3.8 Summary

A particle based axisymmetric simulation of a Hall Thruster plume has been constructed based on the observation that the plume is a quasi-neutral, unmagnetized plasma in which the electrons are effectively collisionless. PIC methods are used to track the motion of charged particles while DSMC methods are used to track a variety of different collisional phenomena including charge exchange, ion-neutral elastic collisions and neutral-neutral elastic collisions. A surface interaction model has also been developed to simulate the presence of solid surfaces and to estimate sputter erosions along an arc centered on the thruster exit. The basic quasi-neutral formulation and thin sheath boundary conditions have been shown to be valid as long as the Debye length is much less than the length scales of objects in the plasma. This is generally the case in all regions directly impacted by the plume, though the simulation may break down in wake regions or in areas very far from the thruster exit. The PIC and DSMC parts of the simulation use a single embedded grid to discretize the domain and resolve areas of high density. An empirical model of an SPT-100 thruster has also been developed and is used to simulate the outlet of a Hall thruster. The model is based on experimental data and contains no free parameters, though the axial ion temperature is varied to match contradictory experimental data. The result is a fast particle simulation capable of simulating higher density plumes than previous possible using particle methods. In addition, the PIC-DSMC method is capable of simulating a relatively wide range of plume conditions, including plasmas where the neutral and ion collision frequencies are the same order of magnitude and plasmas in which neutral-neutral collisions play a significant role. This is a significant advance beyond the capabilities of previous PIC-MCC simulations. The simulation itself was written in ANSI C and run on UNIX workstations. Execution times were typically 2-5 hours on a typical axisymmetric geometry. Because it runs relatively quickly, the axisymmetric plume model was used to verify the basic PIC-DSMC algorithm and the surface interaction model. Results from the axisymmetric simulation and comparisons to experimental data are presented in Chapter 4.

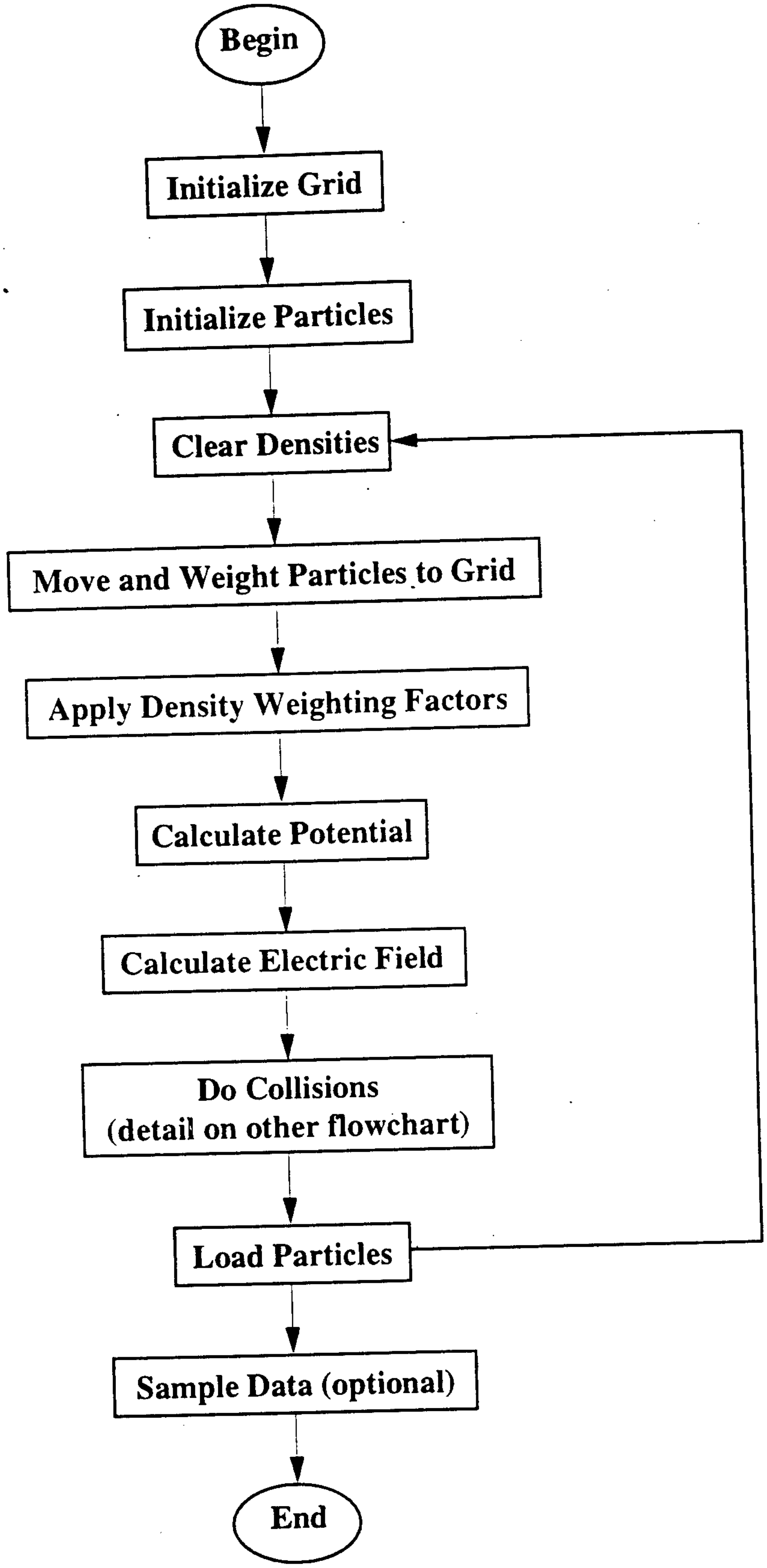


Figure 3.1: PIC-DSMC Simulation Algorithm

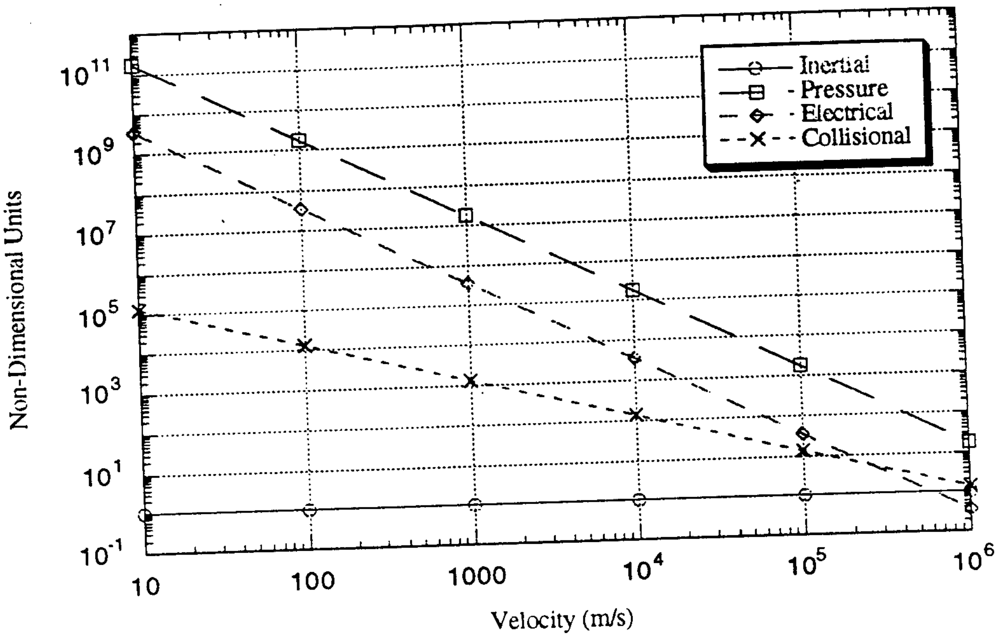


Figure 3.2: Magnitude of Terms in Electron Momentum Equation

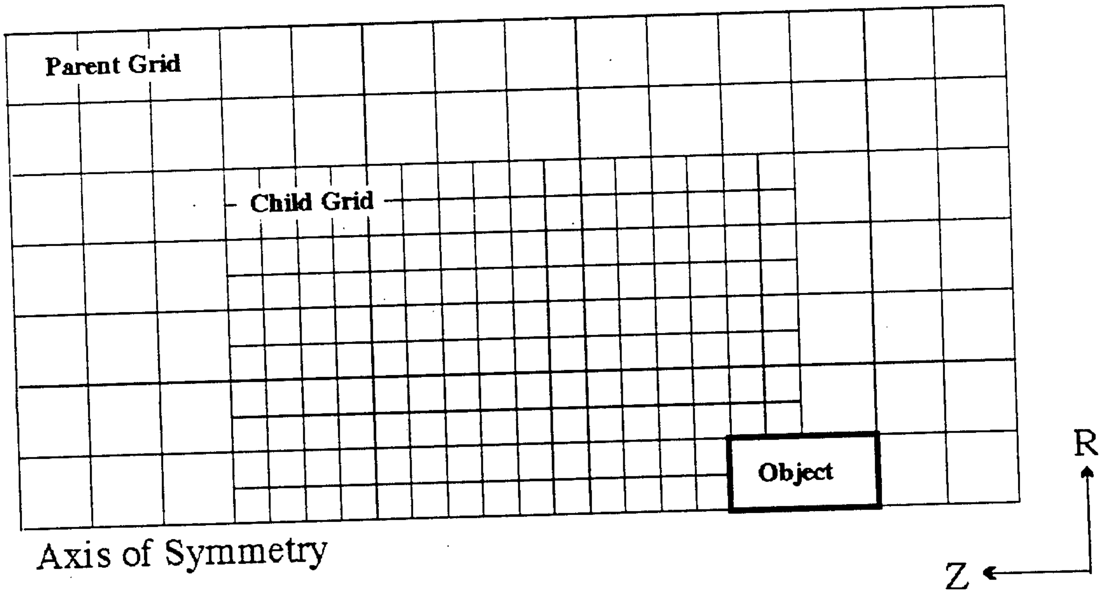


Figure 3.3: Example of an Embedded Grid

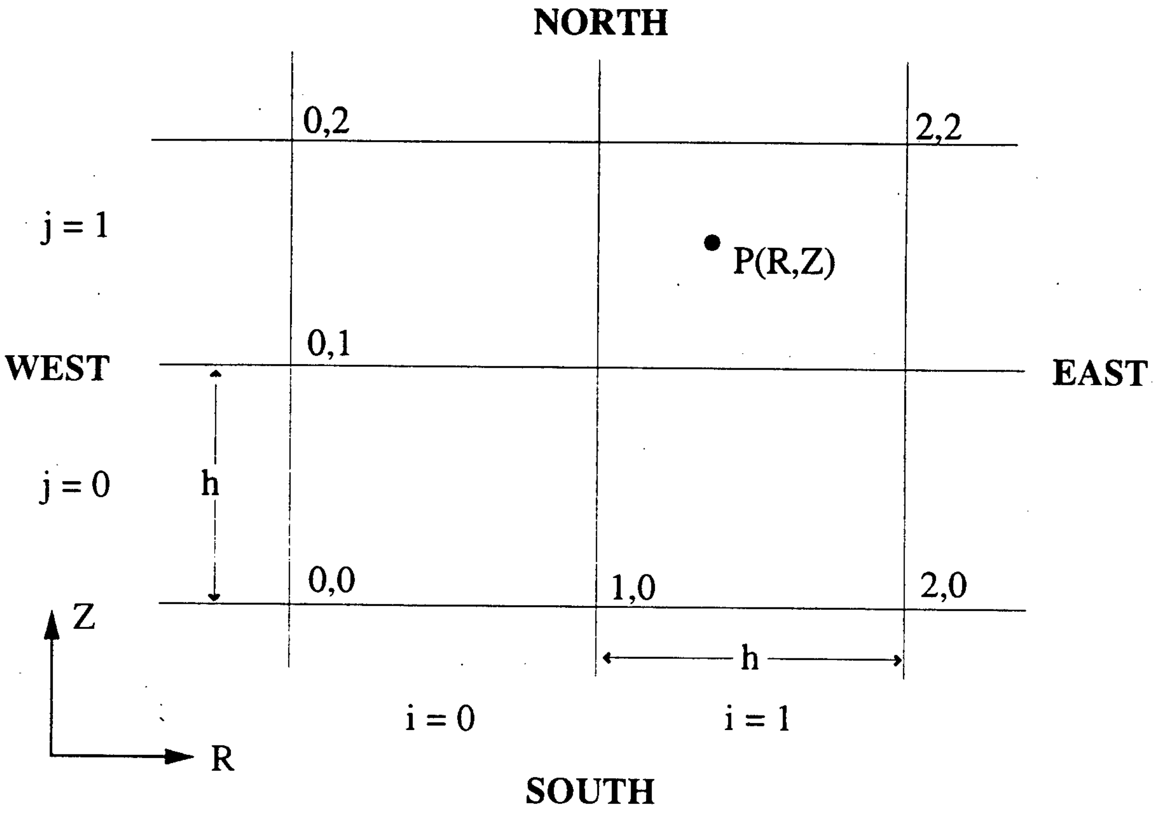


Figure 3.4: Cell, Node and Direction Coordinates

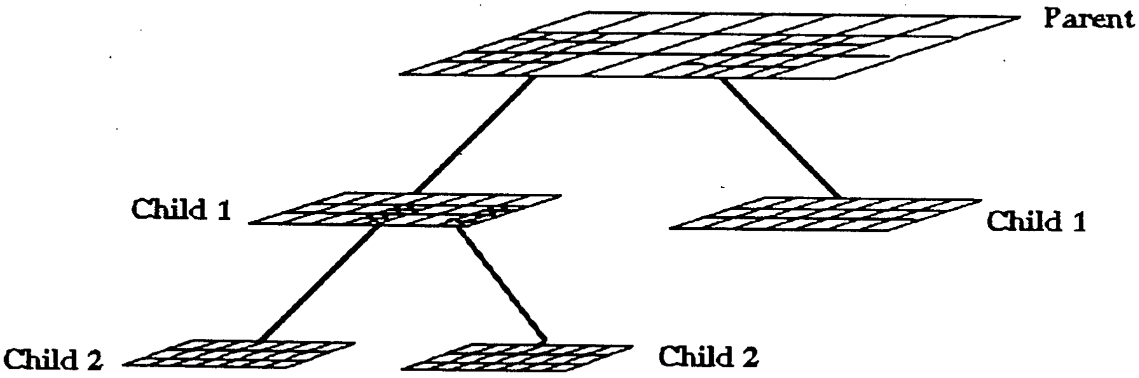
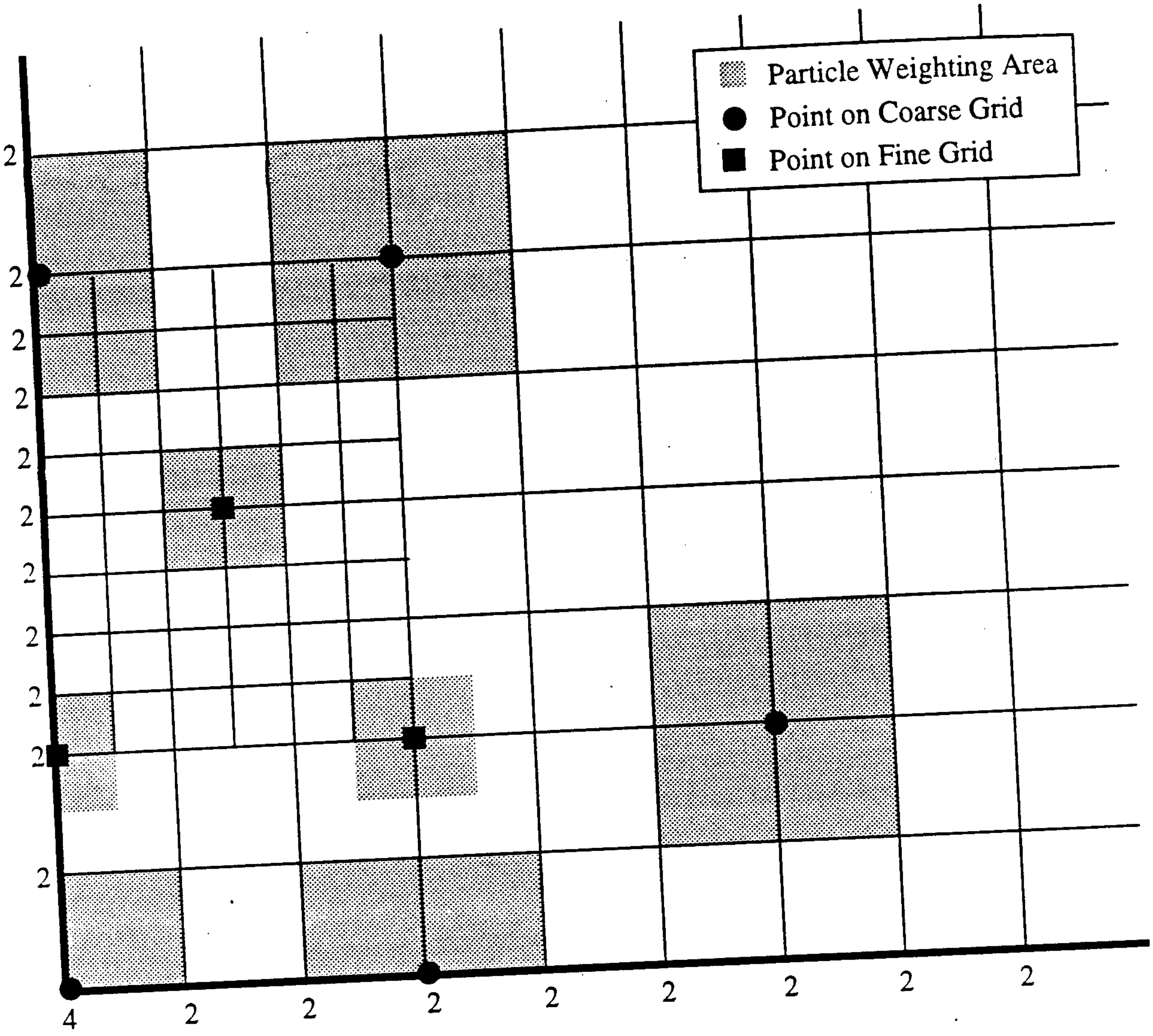


Figure 3.5: Storage of Embedded Meshes in Tree Structure



Numbers indicate nodes with weighting factors not equal to 1

Figure 3.6: Charge Weighting Areas for Nodes on Embedded Grid

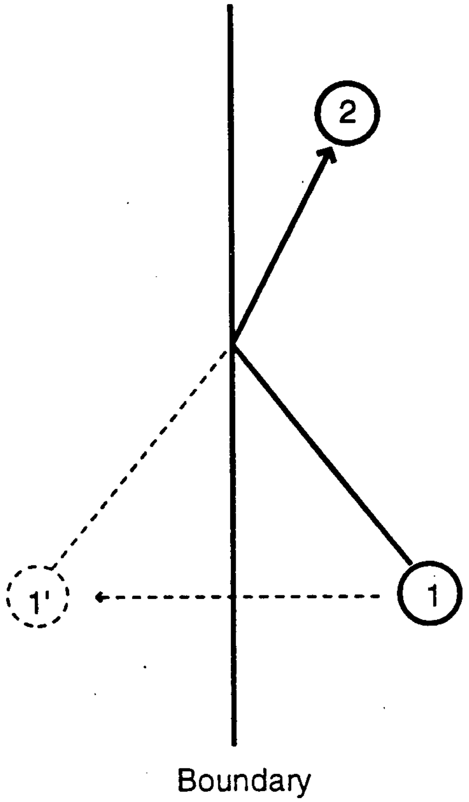
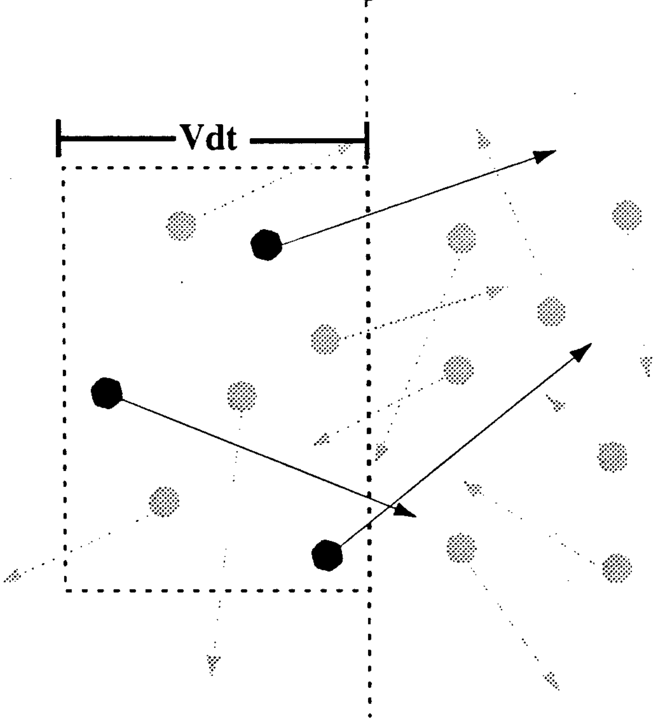


Figure 3.7: Particle Striking Reflective Boundary



One way flux at velocity V
includes only dark particles

Figure 3.8: Planar Cut Through a Maxwellian Gas

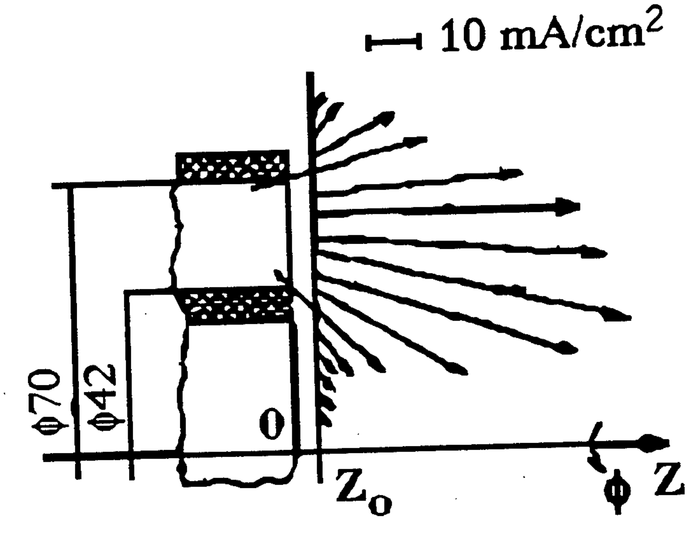


Figure 3.9: Experimental Measurements of Near Field Current Density
From Absalamov [1992] and Gavryushin et al. [1981]

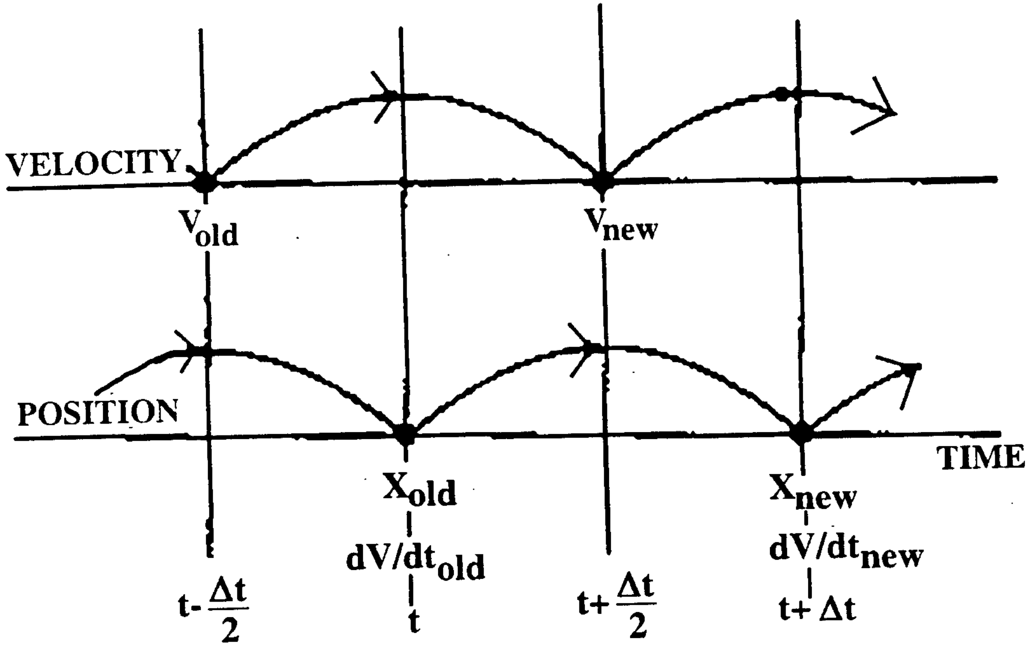


Figure 3.10: Diagram of Leapfrog Method

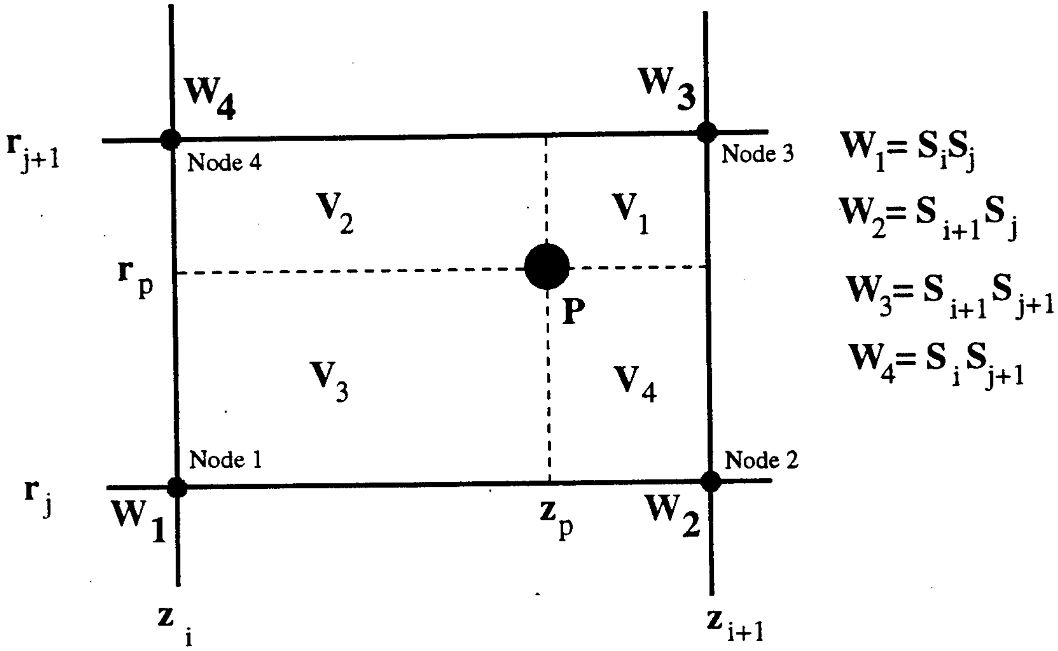


Figure 3.11: Grid Cell Particle Weighting Geometry

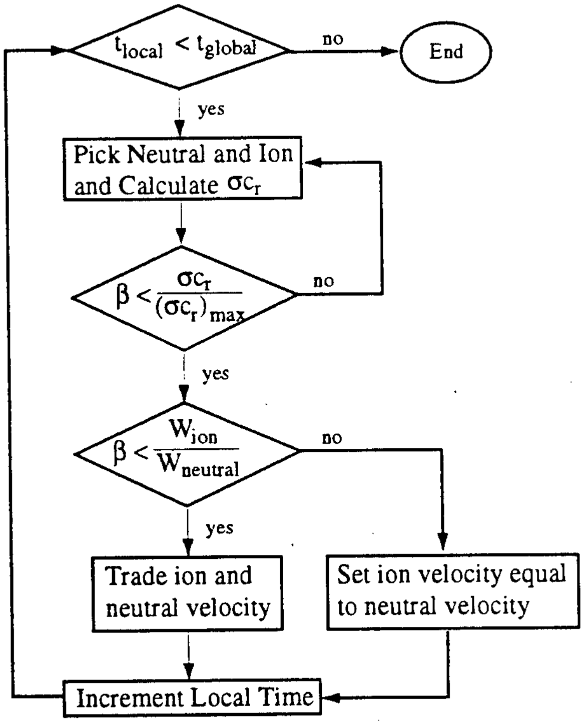

$t_{local} < t_{global}$
no
End
yes
Pick Neutral and Ion and Calculate σc_r
$\beta < \frac{\sigma c_r}{(\sigma c_r)_{max}}$
no
yes
$\beta < \frac{W_{ion}}{W_{neutral}}$
no
yes
Trade ion and neutral velocity
Set ion velocity equal to neutral velocity
Increment Local Time

Figure 3.12: DSMC Collision Algorithm

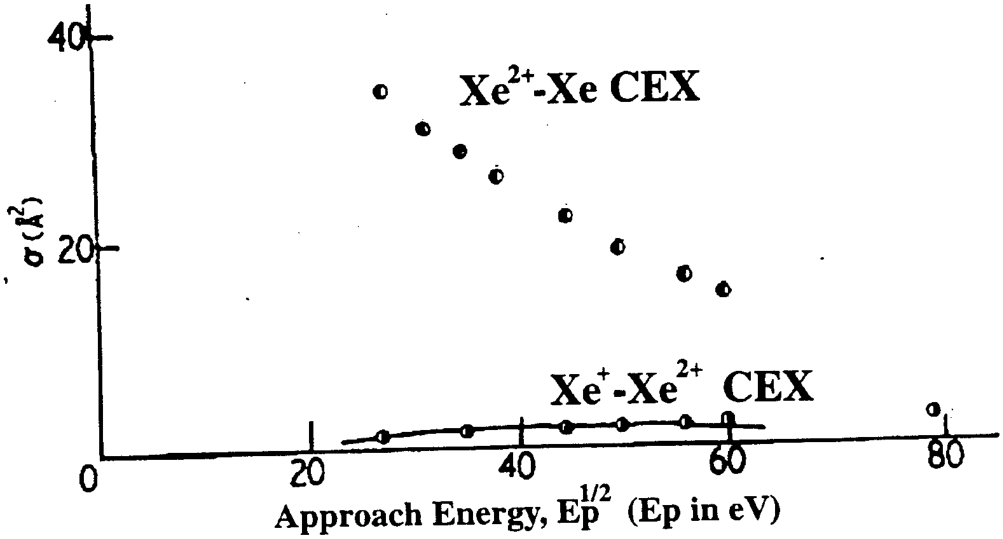


Figure 3.13: Xenon Double Ion Charge Exchange Cross Sections
(Xe^{2+}-Xe CEX is for exchange of 2 electrons)

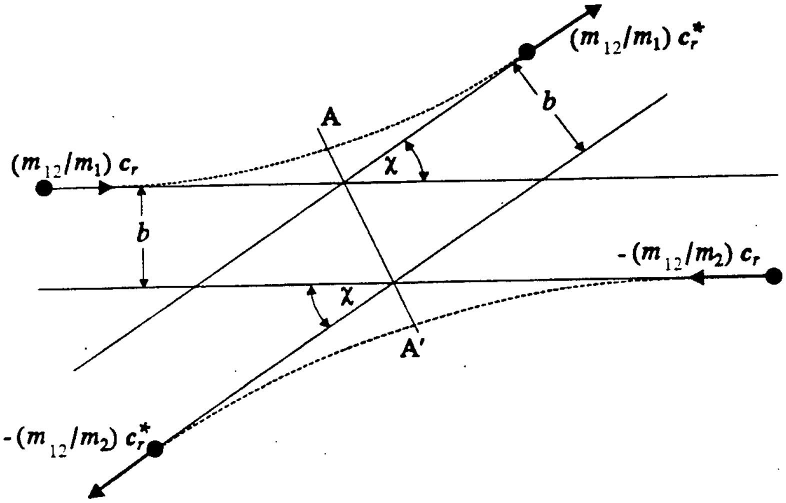


Figure 3.14: Trajectories of Particles Undergoing Collision in COM Coordinates

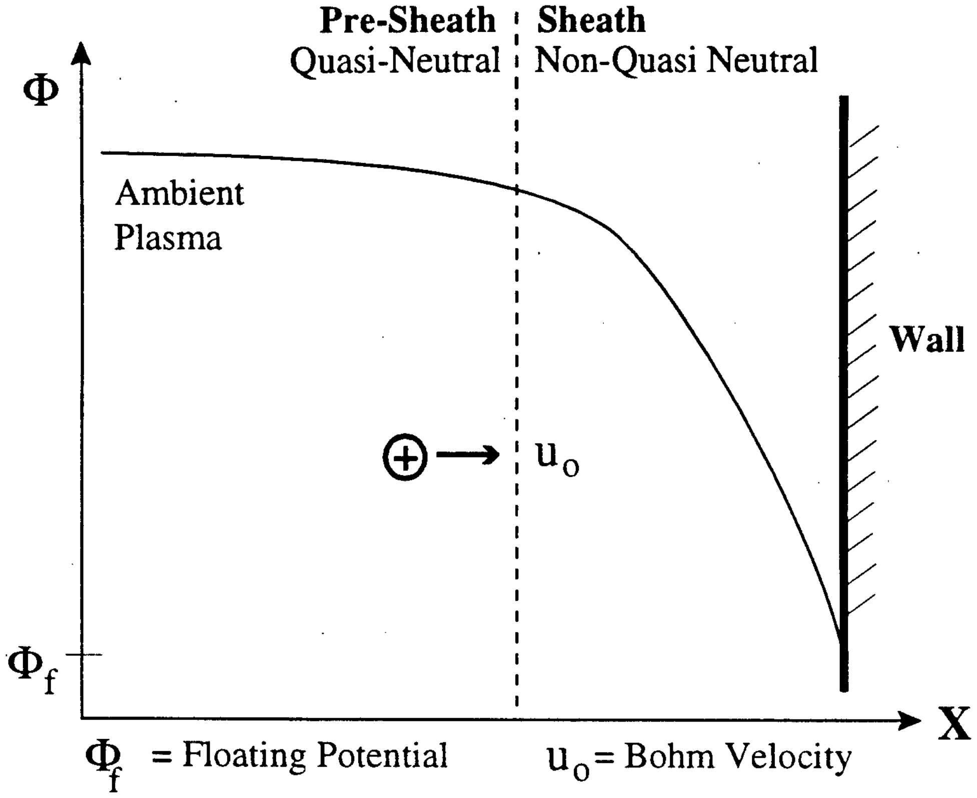


Figure 3.15: One Dimensional Plasma Sheath Structure

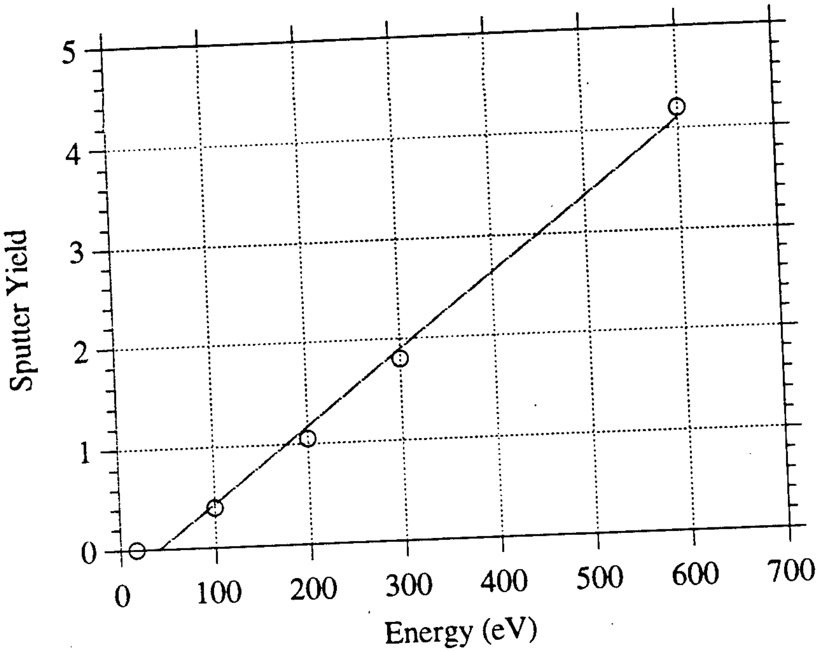


Figure 3.16: Sputtering Coefficient of Silver

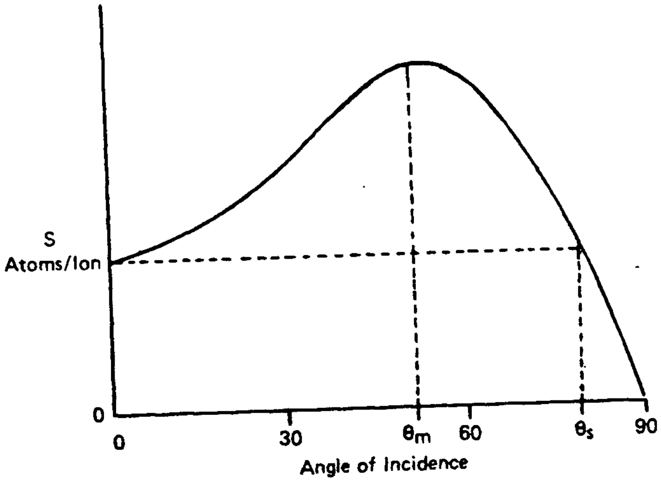


Figure 3.17: Sputtering Coefficient vs. Impact Angle (generalized)
[Chapman pg. 247]

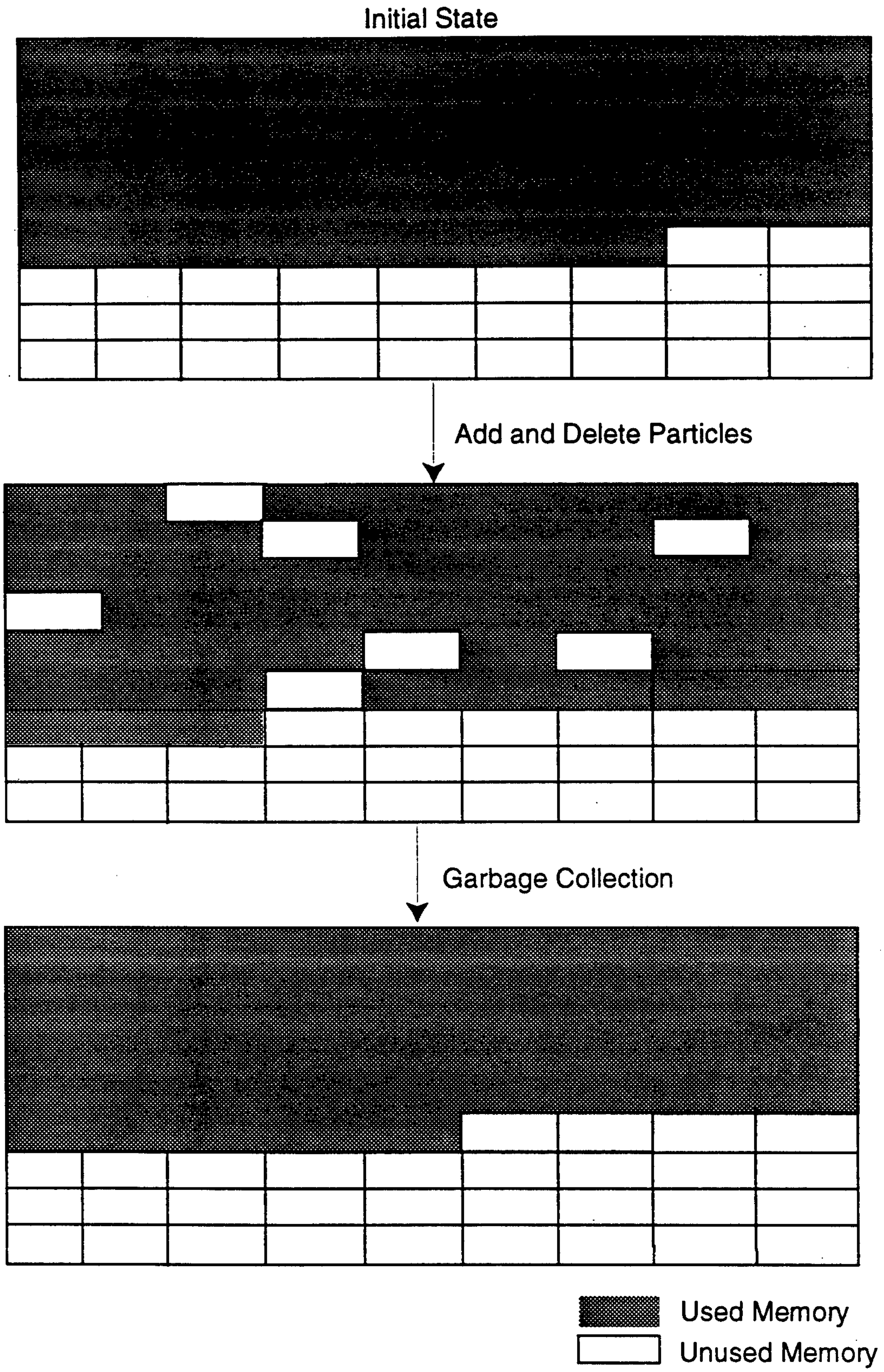


Figure 3.18: Conventional Memory Storage

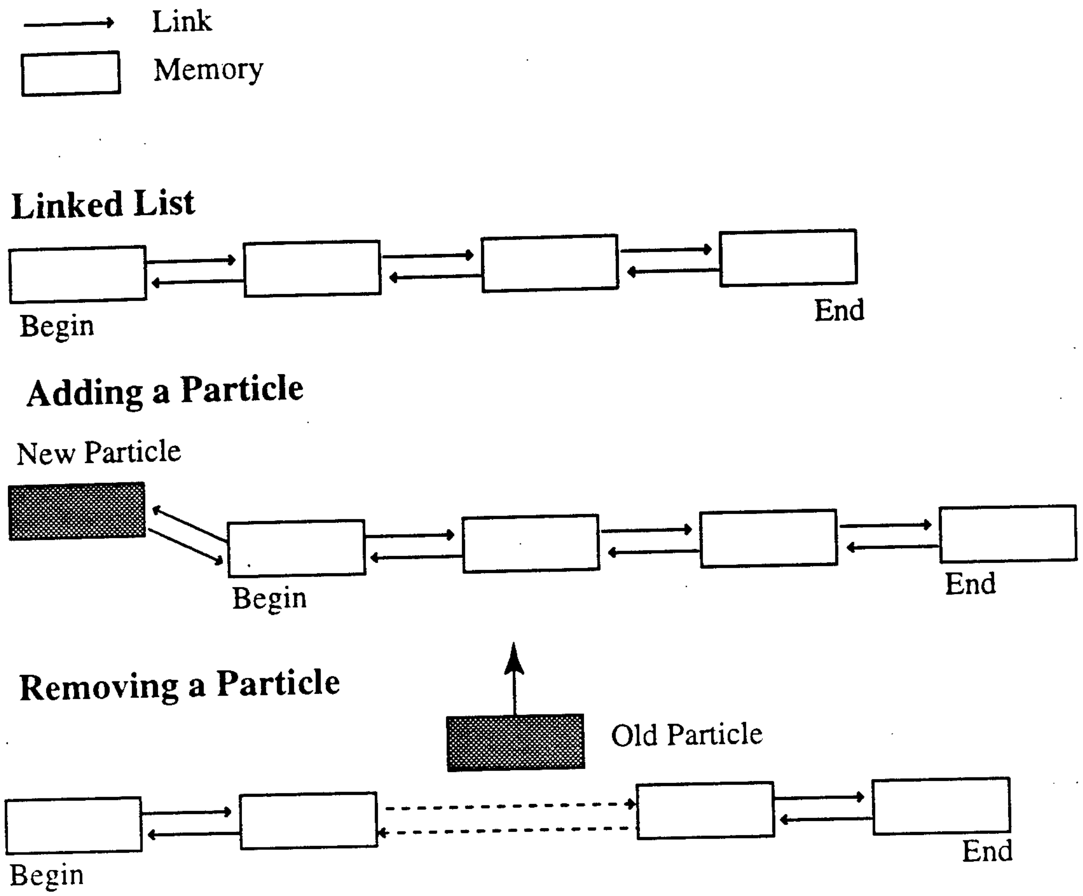


Figure 3.19: Linked List Memory Storage

Chapter 4: Axisymmetric Simulation Results

An axisymmetric PIC-DSMC model of an expanding plasma plume has been constructed and used to simulate the plume of a Hall Thruster. Simulations have been carried out of Hall Thrusters operating in space and on the ground, and comparisons have been made between the results and experimental data. Overall, the model shows good agreement with the experimental data. However, there is significant disagreement between the simulated ion distribution and RPA measurements. We have also found some inconsistencies in existing experimental data, indicating that the data themselves may not be reliable. Where practical, we have attempted to compensate for known errors in the experimental data. This chapter presents various results produced by the axisymmetric PIC-DSMC plume model. Section 1 shows results from various computational and numerical tests which were used to debug the model and verify that a steady state had been achieved. In Section 2, a series of comparisons is made to experimental data taken from the experimental literature. In Section 3, results are presented for an SPT thruster operating under conditions which can not be duplicated in ground experiments and some "rules-of-thumb" are suggested for spacecraft designers planning on integrating Hall thrusters into operational satellites.

4.1 Numerical Checks

In principle, the PIC-DSMC method produces a time accurate plume model. In practice, however, the assumption of quasi-neutrality limits the accuracy of the simulation. When the thruster is turned on at the beginning of each simulation, a plasma plume forms and rapidly expands across the computational domain. Theoretical work has shown that a non-quasi neutral shock wave forms the front of the expanding shock wave, leading to very high expansion velocities and ion wave breaking instabilities [*Crow et al.* 1975, *Sack et al.* 1975]. Since the PIC-DSMC formulation used is quasi-neutral by assumption, these shock waves and transient instabilities are not accurately captured by the simulation. It is therefore important to run the simulation to steady state before using the numerical results. At least two different parameters can be used to determine if a simulation has reached steady state. One commonly used measure is the total number of particles in the simulation. The total number of particles in the simulation becomes constant when the particle flux from the thruster equals the number of particles lost to interior and exterior surfaces. Figure 4.1 shows the total number of particles in a typical simulation changes with time. The timestep is 0.2 normalized units. This timestep is small enough to ensure that the vast majority of the ions in the simulation move no more than one grid cell during each simulated timestep. Figure 4.1 shows that the number of particles in the simulation increases rapidly at first, but reaches a steady state after approximately 2000 iterations. A noticeable amount of numerical variation is present, but the total number of particles varies relatively little after that point. Runs on the same computational domain produce a very similar time history. Figure 4.1 shows that about 2500 iterations are sufficient to assure that a steady state has been reached. Unless otherwise stated, the results presented in this chapter are produced using 2500 iterations with a timestep of 0.2 normalized units.

An important quantity often used to check PIC simulations is the total energy of the particles in the simulation. This quantity is given by the sum of the kinetic and electrostatic potential energy of all of the particles in the simulation, i.e.

$$E_t = \sum_{0}^{N_p} \tfrac{1}{2} m_i \left| v_i^2 \right| + e\phi_i$$

Energy conservation is monitored like mass conservation in computational fluid dynamics codes. In steady state, the total energy in the simulation must be constant. Figure 4.2 shows the total energy from a typical run of the PIC-DSMC plume model. The total energy in the simulation becomes constant (except for numerical noise) after about 500 iterations. This confirms that the PIC-DSMC model conserves energy macroscopically even though DSMC collisions do not necessarily conserve energy at a microscopic level. It also shows that the number of particles in the simulation is a more sensitive measure of steady state than the total energy in the simulation. This is to be expected since the beam ions carry the vast majority of the kinetic energy in the simulation. The fast moving beam ions reach steady state relatively quickly while the slower moving CEX ions take long time to traverse the domain and reach their steady state configuration. The CEX ions carry much less kinetic energy than the beam ions, so the total energy in the simulation is relatively insensitive to the state of the CEX ions. The total number of particles is more sensitive to the CEX ions and is therefore a better measure of the state of the simulation.

Another important factor is the sensitivity of the simulation to the number of particles in it. Ideally, as the number of particles in the simulation increases or decreases, the mean state of the gas (its density, pressure, etc.) should be unaffected. It is important, however, to establish that enough particles are present in the simulation to reach an accurate steady state value. Figure 4.3 shows the measured ion current density through an arc 60 cm from the thruster exit produced using three different particle weighting values. The results show clearly that the number of particles in the simulation has no effect on the final steady state results, even when the number of particles is varied by a factor of four. The results in this chapter were made using the weighting functions corresponding to the smallest of the runs shown in Figure 4.3. Typically, a total of about 100,000 to 200,000 particles are present on the domain when the simulation reaches steady state. This number of particles was chosen because it ensures that about 20 particles per cell are present along the domain centerline. Since the number of particles in each cell scales with the radius squared (at constant density), in general, many more particles are present in cells at the outside edges of the domain.

The test typically used to confirm the accuracy of DSMC models is to simulate a shock wave and look at the width of the shock in units of mean free path. CEX collisions, however, do not produce shock waves. The collision dynamics are relatively simple, so the primary issue with the DSMC model is whether it properly simulates multi-species collision frequencies. To confirm the accuracy of the multi-species DSMC formulation, comparisons were made between a specially modified version of the simulation and a collimated beam of neutral particles entering a uniform background gas. When a collimated beam enters a gas of uniform density, the fraction of the particles which will travel a distance L without undergoing a collision is given by

$$P(L) = \exp\left(-L/\lambda_{mfp}\right) \qquad (4.1\text{-}1)$$

If one ignores particles after they have undergone a single collision, expression (4.1-1) gives the expected change in the beam's density as a function of distance. Expression (4.1-1) is also species independent, so the collision rate for each species should be independent of any other collisions occurring in the gas. Figure 4.4 shows the results of a collimated beam test conducted with two species. The PIC portion of the simulation is not in use because charged particles cause the beam to spread, thus violating the collimated beam assumption. The points show theoretical results based on expression (4.1-1) and the solid lines show results from the simulation. The results for species 2 are noisier than the results for species 1 because they are based on fewer particles, but otherwise the results show excellent agreement with theory. These results show that the DSMC model described in Chapter 3 correctly models the collision rate for multiple species undergoing collisions in a mixed environment.

Time averaging has been used to reduce the noise level in many of the computational results presented in this chapter. Instead of presenting the instantaneous ion density, for instance, we use the average ion density from every other timestep as tabulated over the last 500 timesteps (a total of 250 sample points). In steady state, the ergodic principle ensures that the density obtained by time averaging will match the density which would have been obtained by an ensemble average at the same point in space. Figures 4.5 and 4.6 show time averaged and non-time averaged contour plots of the ion density. The use of time averaging produces a plot with less noise without affecting the overall steady state values. An alternative to the use of time averaging is the use of additional particles, but both the computation time and memory required by the simulation scale linearly with the number of particles in the simulation. An upper limit is defined by dividing the amount of RAM available by the memory taken up by each particle. In our implementation, this is 64 bytes. On a workstation with 128 Megabytes of RAM, the absolute upper limit on total number of particles is 2,000,000. The actual number of particles will be much less due to system overhead and the memory for the grid. To improve computation time, we typically use about 200,000 particles in our simulation.

Finally, one numerical issue which needs to be addressed is that of uniqueness. Since the system we are modeling is non-linear, it may (in principle) be possible to produce more than one solution that fits a given set of boundary conditions. Our model is not strictly time accurate because it does not include non-quasi-neutral phenomena. It is therefore possible that the steady state solutions produced by the model might correspond to non-physical solutions to the plume interaction problem. For example, the plumes of plasma contactors have been shown to contain non-quasi-neutral double layers which act as interfaces between the main plume and the ambient plasma. These regions would not appear in our models because the plume is quasi-neutral by assumption. Nevertheless, the solutions produced by the model may appear self-consistent and may match the boundary conditions imposed on the problem. It is not clear how one would rigorously demonstrate that our solutions are physical and/or unique. We note that experimenters have not observed the presence of shocks, double layers, or other phenomena typically associated with the presence of a bifurcation in the solution. In addition, we have made a thorough set of comparisons to experimental data to confirm the basic accuracy of our results and show that they produce the right qualitative and quantitative behavior. It therefore seems unlikely that the simulation could produce non-physical solutions to the plume interaction problem. We acknowledge, however, that uniqueness has not been rigorously shown and will not be further addressed in this work.

In addition to the various tests cited above, there are several further efforts which could be made to verify the quasi-neutral PIC-DSMC method against theory. A supercomputer, for instance, could be used to run non-quasi-neutral PIC-DSMC simulations. These results could be compared to results from the quasi-neutral PIC-DSMC simulation to see if the analytical source model correctly calculates sheath potentials and to confirm that non quasi-neutral effects in wake regions have no effect on the overall structure of the plume. An additional verification test which has yet been conducted is to compare simulated results with a known theoretical test case. We have been unable to construct a test case which fits the assumptions of the simulation and has a known analytic solution. It would, however, be useful to define such a case and further verify the quasi-neutral PIC-DSMC model.

4.2 Comparisons to Experimental Data

While numerical tests help check the internal consistency of a simulation, an experimental comparison is required to verify that a computer model accurately represents reality. Fortunately, a variety of experimental efforts have been undertaken to study the plumes of SPT thrusters. The axisymmetric PIC-DSMC plume model was partially verified against experimental data taken from the literature. One issue with the literature data is the lack of error bars, which makes comparisons difficult. Additional data were provided specifically for this work by the researchers at the University of Michigan. Comparisons were made to verify the computational simulation, examine the system's sensitivity to axial ion temperature, and to develop an understanding of the influence that the background neutrals present in vacuum tanks have on experimental results. A single computational domain was used for all of the axisymmetric simulations and is shown in Figure 4.7. The domain has dimensions of 1.4 m x 1.1 m and represents an SPT-100 thruster operating in the presence of a background neutral gas. The presence of background gas was simulated by creating a neutral Xenon background of particles at a uniform density in a Maxwellian distribution and by including thermal Xe fluxes along exterior boundaries (as discussed in section 3.7). This section presents several comparisons between data and the simulation and discusses their various implications. Section 4.2.1 describes simulations of the SPT thruster operating with different axial ion temperatures used in the source model. Sections 4.2.2-4.2.4 provide further comparisons between experimental data and results from the simulation.

4.2.1 Sensitivity to Axial Ion Temperature

As discussed in section 1.1.3, laser induced fluorescence measurements by Manzella indicate that the temperatures of ions in the plume in the axial direction is approximately 3.4 eV. This value appears at first glance to contradict the RPA measurements of the ion velocity distribution shown in Figure 2.2 and in Figures 4.27-4.29. As will be discussed in section 4.2.4, however, the RPA measurements are actually consistent with the LIF measurements. However, the apparent contradiction led us to examine the sensitivity of the overall simulation to the axial ion temperature used in the source model. The axial ion temperature was varied parametrically in a series of simulations of the SPT-100 thruster. Runs were conducted simulating an SPT-100 operating at a background pressure of 2.2 x 10^{-6} Torr. The axial ion temperature was set equal to three different values in these simulations: 3.4 eV, 15 eV, and 34 eV. Figure 4.8 shows simulated results with an axial temperature of 3.4 eV and Figure 4.9 shows simulated results with a temperature of 34 eV. Figures 4.8 and 4.9 are both contour plots of the potential overlaid with vectors showing ion current direction and magnitude. Larger vectors correspond to higher current magnitudes. In both cases, the primary beam is visible as a region of high potential near the centerline and backflow current is clearly visible to the sides and rear of the thruster. However, in Figure 4.8, a bump is visible in the potential contours to the front and side of the thruster. This bump is caused by CEX ions and has been observed experimentally in the plumes of ion thrusters [*Kaufman* 1975 as discussed in *Samanta Roy* 1995]. In Figure 4.9, however, no such bump is visible. These results indicate that some aspects of the plume's structure are very sensitive to the axial ion temperature. But since experimenters have not yet measured the potential throughout the plume region, it is not clear from these figures which ion temperature gives the correct plume structure.

One quantity which has been well documented by experimentalists is the ion current density. Figure 4.10 compares an experimental measurement of the ion current density at a pressure of 2.2×10^{-6} Torr to a simulation made with an axial ion temperature of 3.4 eV at a background pressure of 2.2×10^{-6} Torr. This temperature matches LIF measurements made by Manzella [1994]. The current flow to the thruster has been enhanced to account for current from the ambient plasma, as discussed in section 4.2.2 below. The potential bump present Figure 4.8 is visible in the simulated results shown in Figure 4.10 as a bump in the ion current measurements. This bump is clearly missing from the experimental data. Otherwise, the two sets of results agree fairly well, to within about a factor of 2-3 across most of the measured range. Figure 4.11 compares the results of simulations taken at three different axial ion temperatures with experimental measurements taken at a pressure of 5.6×10^{-6} Torr. Again, a wing structure is clearly visible in the simulated results when the axial ion temperature is 3.4 eV. However, as the axial ion temperature rises, the bump disappears and the simulation shows marginally better agreement with the experimental data. From Figure 4.11, it is clear that higher axial ion temperatures improve the match between the simulation and the experimental measurements. The reason for this interaction is not immediately obvious and merits some discussion.

The presence of the wing structure in Figures 4.8 and 4.10 is related to the turning radius of ions leaving the acceleration channel. As the plume expands, the density gradient at the edge of the plume creates an electric field that turns ions away from the centerline at a rate which is related to their initial velocity. This gradient is clearly visible in Figures 4.8 and 4.9 as tightly grouped potential contours located to the side and front of the thruster. Ions with relatively high velocities have large turning radii and tend to follow straight trajectories, while ions with low velocities have small turning radii and are quickly turned towards the edge of the plume. This effect can be see clearly in Figures 4.12 and 4.13. Figure 4.12 shows the trajectory of beam ions in the plume region. These ions are clustered along the centerline and follow relatively straight trajectories. Figure 4.13 shows the trajectories of CEX ions in the plume region. Unlike the beam ions, the CEX ions tend to follow curved trajectories and are pushed to the side and rear of the thrusters. These ions form the potential wing visible in Figure 4.8.

As was noted above, the presence of wings is sometimes seen experimentally with ion thrusters. In this case, the ions in the main beam have a low axial temperature and are moving in a small range of velocities which is much faster than ions created by CEX. As a result, the CEX ions exit to the sides of the plume and form a "wing" like structure similar to that shown in Figure 4.8. In a Hall thruster, the beam as a whole is also moving much faster than the CEX ions. But because the axial ion temperature is relatively high, the beam ions span a wide range of velocities. As a result, different parts of the beam turn at different rates, producing a much less defined beam boundary. The slower ions at the edge of the beam end up merging with the CEX ion wings to create a relatively smooth density variation. As a result, the potential "bump" seen in Figure 4.8 disappears when the axial ion temperature rises, as shown in Figure 4.9 The presence of wings is therefore highly dependent on the axial ion temperature. If the ion temperature is low, the beam is relatively tight, and the CEX wings are clearly visible. If the ion temperature is high, the beam is relatively diffuse, and the CEX wings are absorbed into the diffuse beam structure.

Figure 4.11 shows that the higher ion temperatures show better qualitative agreement with experimental data. We therefore used 34 eV as the axial ion temperature in our SPT-100 plasma source model. This value is used for all axisymmetric and three dimension results presented in the rest of this document unless otherwise stated.

There are several conclusions which can be drawn from this discussion. First, we conclude that the relatively tight plumes associated with ion thrusters are a function of their relatively low axial ion temperature. Second, we conclude that higher axial ion temperatures tend to lead to spreading of the ion beam. Third, we conclude that some problems must still exist in our beam ion source model. Although we can duplicate the measured ion current distribution, we require the use of a higher ion temperature than is actually present in the thruster. This high temperature is not physically accurate, so we fail to duplicate the detailed ion velocity distribution. This point will be discussed further in the section comparing the model to RPA measurements (Section 4.2.4).

4.2.2 Ion Current Density

In order to verify the axisymmetric plume model, a series of comparisons were made to experimental work by *Manzella* [1995]. Manzella used circular molybdenum probes to measure the ion current density in an arc 60 cm. from the exit of an SPT-100 thruster. His measurements span a range of 200 degrees and include measurements taken in the thruster's backflow current region. The tests were designed to characterize the plume and measure the test facility's influence on the thruster. In one set of tests, Manzella varied the facility's background pressure by over an order of magnitude while measuring the ion current density profile. These tests provide ideal data for verifying the PIC-DSMC method. Since the ion collision frequency varies with the background pressure, data collected across a range of background pressures helps verify both the PIC and DSMC sides of the simulation.

A series of simulations were run to examine the effect that the background pressure present in ground test facilities has on the structure of the SPT-100 plume. Four different ambient pressures were simulated, each corresponding to an experimental result reported by *Manzella* [1995]. These results are summarized in Figure 4.14. The presence of background gas was simulated by adding neutral Xenon particles in a Maxwellian distribution and by including thermal Xenon fluxes along boundaries with the ambient plasma (as shown in Figure 4.7). The background ion temperature was assumed to be 300 K (room temperature) and the background number density was determined using the ideal gas law, as shown in Table 4.1.

Pressure	Number Density
2.2 x 10^{-6} Torr	7.085 x 10^{16} m^{-3}
5.6 x 10^{-6} Torr	1.803 x 10^{17} m^{-3}
2.5 x 10^{-5} Torr	8.050 x 10^{17} m^{-3}
6.3 x 10^{-5} Torr	2.029 x 10^{18} m^{-3}

Table 4.1: Simulated Pressures and Number Densities

An anode propellant flow rate of 4.99 mg./sec. and a cathode flow rate of 0.38 mg./sec. were used in these simulations. This is consistent with flow rates used by Manzella in his experiments. The ion current density was measured by counting the number of ions crossing a virtual surface placed in an arc 60 cm. from the anode exit. A set of simulated results is presented in Figures 4.15 and 4.16 along with comparisons to experimental data. The plots correspond to ambient pressures of 2.5 x 10^{-5} Torr and 2.2 x 10^{-6} Torr respectively. Manzella's data has been shifted to the left by three degrees in order to line up the peaks of the ion current density plots. Although the same basic trend are present in both sets of data, there are notable differences in the magnitude of the two results. The computed results consistently underpredict the current density at most pressures and angles. As a result, the total beam current is much larger in the experimental case than it is in the simulation. This results is surprising since the total propellant flow is the same in both cases. The total beam current can be calculated numerically by integrating the data presented in Figure 4.14. The results are shown in Table 4.2. The thruster's discharge current is an upper bound on the amount of current which can be carried by the ion beam. The beam current will actually be less than the discharge current because some electrons will flow back to the anode instead of neutralizing the main beam.

Pressure (Torr)	Integrated Current (A)	Compensation Fraction
2.2 x 10^{-6}	4.69	1.124
5.6 x 10^{-6}	5.26	1.260
2.5 x 10^{-5}	5.97	1.431
6.3 x 10^{-5}	6.15	1.474
Discharge Current	~4.5	N/A

Table 4.2: Experimentally Measured Beam Current vs. Pressure

Table 4.2 shows that the experimentally measured beam current consistently exceeds the discharge current by between 4% and 35%. This is physically impossible and indicates that some error is present in the experimental data The amount by which the beam current exceeds the discharge current varies with pressure, suggesting that some interaction is occurring between the thruster and the ambient background in the tank. The excess current has been observed by Russian researchers and is thought to be due to the formation of a background plasma (rather than neutral) gas in the vacuum tank. The background plasma adds a thermal contribution to the measurements of the ion current and can also recirculate within the tank, creating additional external current flow.

Unfortunately, no measurements have been taken of the background plasma in ground experiments and its contribution to measured ion currents. We therefore compensate for the extra current flow numerically by adding an additional propellant to the flow going through our numerical thruster. This magnitude of the compensation flow was determined using the following procedure. The nominal current flow through the anode is given by

$$I_b = \frac{e\dot{m}\eta_i}{m_{XE}}(\eta_{++} + 1)$$

Where $\dot{m}$ is the total mass flow through the anode, η_i is the overall ionization fraction, and η_{++} is the fraction of the plasma which consists of double ions. Based on experimental measurements, our SPT-100 source model uses a total ionization fraction is 0.95 and a double ion fraction of 0.2 (as discussed in section 3.3.3). At an anode propellant flow rate of 4.99 mg./sec., the beam current is therefore 4.173 Amps. The magnitude of the compensation flow was determined by multiplying the nominal propellant flow by the following fraction

$$f = \frac{I_m}{I_b} \tag{4.2-1}$$

Where I_m is the actual measured current flow as shown in Table 4.2. The resulting current compensation fractions are also shown in the third column of Table 4.2.

When this compensating propellant flow is included in the simulation, the results compare somewhat more favorably to the experimental data. Figure 4.17 shows the simulated ion current density as measured at four different ambient pressures in an arc 60 cm from the thruster exit. For comparison, experimental measurements by Manzella are shown in figure 4.14. Figures 4.18-4.22 show individual comparisons between the simulated and experimental results at each of the four pressures measured in the experiment. The simulated results show fair agreement with experimental data, and the same shapes and trends are present in both sets of results. In addition, the results agree a little better numerically and are within a factor of 2 to 3 across most of the domain, with larger disagreement at very high and very low angles. The error on the ion current measurements is unknown, but measurements of this type are typically accurate to within about 50% [*Gallimore*, Personal Communication, 1996]. In view of the uncertainty, it can be said that the PIC-DSMC model agrees well with experimental data, even as the ambient pressure is varied by an order of magnitude. The simulated results do tend to overpredict the current along the centerline and underpredict it at high angles. The overprediction along the centerline may be a result of numerical artifacts along the centerline, inaccuracies in the source model, or saturation of the ion current probe during the experiments. Numerical artifacts come about because the current density is determined by dividing a numerically measured flux of particles by the sample area. In an axisymmetric geometry, both the number of particles and the sample area scale as $1/r^2$. Therefore, near the centerline, a small number of particles is divided by a very small area to produce the measured current densities. This procedure is obviously sensitive to small sampling errors and can produce numerical artifacts. The underprediction at high angles may be a result of errors in measured CEX cross sections or inaccuracies in the source model. The estimated error in the CEX cross sections used in our simulation is 20% [*Hasted* 1964]. We note also that there is a great deal of uncertainty in the experimental measurements. The pressure measurements in particular are thought to be accurate to within only half a decade [*Manzella*, Personal Communications, 1996]. If the pressure reported is consistently lower than the actual pressure in the tank, for instance, the simulation would consistently underpredict the backflow current at the edges of the plume. In view of the experimental uncertainty, we believe that the experimental comparison to Manzella's data partially verifies the accuracy of the axisymmetric PIC-DSMC formulation.

To further verify the PIC-DSMC plume model, additional comparisons were made to ion current data provided by Colleen Marrese and Alec Gallimore at the University of Michigan. Marrese and Gallimore used a Faraday Probe to measure the ion current density along an arc 50 cm. from the exit of an SPT-100 thruster. The measured pressure in the tank was 5.2 x 10^{-5} Torr. The measured ion current was integrated to determine the total beam current measured in the experiment. The result is 3.55 Amps, or 0.62 Amps less than the beam current given by equation (4.2-1), so no compensation currents are necessary. Otherwise, simulations were conducted using the same techniques described earlier in this section. Figure 4.22 shows the ion current density measurements made at the University of Michigan and the results of our simulation. The error bars on the results are again probably about 50%. The results show excellent agreement. The simulated resulting are easily within 50% of the experimental results across all of the domain except for a small region near the centerline. The disagreement at the centerline could be due to problems with the source model or due to numerical artifacts (as discussed in the comparison to Manzella's data above). Overall, however, these results serve to partially verify the PIC-DSMC plume model.

4.2.3 Erosion Rates

The results presented in section 4.2.2 serve to partially verify the overall plume model, they do not verify the techniques used to model surface interaction and sputtering effects. These models were verified independently using experimental results from *Randolph et al.*[1994]. Randolph and Pencil placed samples of silver foil, solar cell interconnectors, and solar cell coverglasses in a vacuum tank and exposed them to the SPT-100 plume for a period of 200 hours. The samples were placed on witness plates which were mounted in an arc at a distance of 1 m from the anode exit. The plates faced towards the thruster to the samples and were normal to the expected flow. These tests were conducted at a pressure of 3 x 10^{-5} Torr and collimators were used to limit the samples' field of view. The collimators served to shield the samples from contamination from the walls of the tank itself. Figure 4.23 shows simulated results for the erosion of silver after two hundred hours of exposure at a pressure of 2.5 x 10^{-5} Torr. The results of Randolph and Pencil are superimposed on the simulated results. In both cases, the measurements are taken 1 m from the anode exit. Although the simulated results are noisy, they agree fairly well with the experimental data. The same shapes and trends are clearly present in both results, though the simulation consistently underpredicts the measured erosion, sometimes by more than a factor of 2. Figure 4.24 shows similar results for the erosion of Quartz after 200 hours of exposure to the plume. Superimposed on these results is the experimentally measured erosion of solar cell coverglasses after 200 hours of exposure to the plume. Although the same trends are apparent, the agreement is not as good, and the simulation consistently underpredicts the data by a factor of three or more. The relatively high quality of the results for silver suggests that the source of the inaccuracy is probably in the relationship used for the sputtering coefficient. The sputtering coefficients used for the solar cell coverglasses represent a generic "quartz glass". The solar cells themselves may use glass with a different composition and lined are with an anti-reflective coating. The sputtering coefficients of solar cell coverglasses has not been reported in the literature and is often unknown. When it is known, the information is held privately and considered proprietary. The material used in the solar cell coverglasses probably has a higher sputtering coefficient than the "quartz glass" modeled in the simulation. In addition, the sputtering coefficients used in the simulation are valid for ions striking normal to the surface. In reality, however, ions which approach at high angles may have sputtering coefficients several times higher than those used in the simulation [*Chapman* pg. 247]. Unfortunately, no data are available on the angular dependence of the sputtering coefficient of Xenon atoms on Silver or Quartz surfaces. Such data are required to further refine the sputtering model.

Randolph et al. [1994] developed an empirical model to predict the erosion of interconnectors and solar cell coverglasses on surfaces exposed to the plume of an SPT-100 thruster. Their model is of the general form

$$j = \frac{R_o^2}{r^2}\left(a_o + \frac{a_1}{\theta^2 + a_2}\right)$$

Where j is the ion current density (mA/cm^2), R is a reference distance of 1 m, r is the downstream distance (m), θ is the divergence angle (°), $a_o = 0.0014895$, $a_1 = 103.12$, and $a_2 = 60.169$. Figure 4.25 shows Randolph's prediction of the erosion rate for silver compared to his own experimental measurements and the PIC-DSMC model. Randolph's generally predicts erosions rates which are higher than the measured rates. His model shows "good agreement with the sample measurements at angles less than 50°", but less good agreement beyond 50° [*Randolph et al.* 1994]. Our model tends to predict erosion rates which are lower than the measured rates, but which agree to about the same level of accuracy as Randolph's model. Our model also contains more computational noise, which causes our predictions to wander more than Randolph's model. At a divergence angle of 90 degrees, neither model gives very good agreement. In this range, however, the erosion rate is small enough to be of little interest to designers. Nevertheless, it should be noted that our model gives a conservative estimate of erosion rates.

4.2.4 RPA Measurements

Retarding potential analyzers (RPA's) have been used to measure the energy distribution of ions in the plume. Because the PIC-DSMC algorithm is a particle based method, an RPA can be simulated by recording the energy of individual macroparticles crossing a virtual surface. Over time, this produces a picture of the one-way ion distribution function, similar to that produced by an RPA in the laboratory. Comparisons have been made between two different sets of RPA data and results from the axisymmetric simulation. This section shows the results of these comparisons and describes their implications for the simulation.

Figure 4.26 shows a simulated RPA distribution at a chamber pressure of 6.3 x 10^{-5} Torr. Figure 4.26 was constructed by recording the energy of particles crossing an arc 60 cm from the anode exit. Particle fluxes were recorded in increments of 0.5 degrees, so the curve marked "7.5 degrees" corresponds to ions crossing at angles between 7.25 and 7.75 degrees from the centerline. Figure 4.27 shows experimental RPA measurements made along an arc 1 m away from the thruster at a pressure between 5 and 6 x 10^{-5} Torr [*Absalamov* 1992]. Figures 4.28 and 4.29 show experimental RPA measurements made by Marrese and Gallimore [1996] at the University of Michigan along an arc 0.5 m from the thruster at a pressure of 5.2 x 10^{-5} Torr. The Michigan data has been corrected for the local plasma potential by subtracting 14 V from the energy levels measured in the experiment. The width of the peak in the experimental data is quite large. In Figure 4.28 the peak has a half width of about 50 volts. When Manzella measured the axial ion temperature using LIF techniques, he reported a temperature of about 3.4 eV. Although the width of the peaks in the RPA data appear to contradict this result, the two results are actually consistent with each other. This point bears some discussion.

Temperature is defined as the standard distribution of the ion velocities around a their mean velocity (the drift velocity). The ion temperature can be estimated from the RPA measurements by assuming that the ions have a Gaussian (Maxwellian) distribution. In 1-D, the Gaussian distribution function is given by

$$f(v_x) = \left(\frac{m}{2\pi kT_i}\right)\exp\left(-\frac{m}{2kT_i}(v_x - v_d)^2\right)$$

Where T_i is the ion temperature, m is the ion mass, v_x is the particle velocity in this dimension and v_d is the overall drift velocity. Plotting $f(v_x)$ as a function of v_x results in a single, symmetric peak centered on the drift velocity. The width of the peak is entirely determined by the ion temperature. The drift velocity serves only to shift the center of the distribution function. The RPA data in Figures 4.27-4.29 shows the ion distribution function plotted against energy rather than velocity. When the Gaussian distribution function is written as a function of energy, it takes on the following form.

$$f(\varepsilon) = \left(\frac{m}{2\pi kT_i}\right)\exp\left[-\left(\frac{\sqrt{\varepsilon} - \sqrt{E_d}}{\sqrt{T_{ev}}}\right)^2\right]$$

$$\varepsilon \equiv \frac{\frac{1}{2}mv_x^2}{\cdot e}$$

$$E_d \equiv \frac{\frac{1}{2}mv_d^2}{e}$$

Plotting f(ε) vs. ε results in a single non-symmetric peak whose width depends on the temperature and on the drift velocity. At high drift velocities, the apparent width of the distribution function is much larger than the temperature. In Figure 4.27, for instance, the peak of the distribution function is at an energy of 280 Volts. We will assume that this is drift energy, i.e. E_d = 280 eV. If the ion distribution is a Gaussian distribution, when

$$\frac{\sqrt{\varepsilon} - \sqrt{E_d}}{\sqrt{T_{ev}}} = 1$$

the height of the distribution function should be 0.36787 times its peak value. In Figure 4.27, the distribution function for the centerline RPA measurement is 0.36787 times its peak value when the collector potential is 305 Volts and 225 Volts. This results in two different axial ion temperatures

$$T_{ev} = \left(\sqrt{305} - \sqrt{280}\right)^2 = 0.534 \text{ eV} \qquad \text{(high value)}$$

$$T_{ev} = \left(\sqrt{225} - \sqrt{280}\right)^2 = 3.0 \text{ eV} \qquad \text{(low value)}$$

If the ion distribution were a true Gaussian, the temperatures would be the same. In this case, however, a Gaussian is not a particularly good approximation for the ion distribution. Nevertheless, both temperatures are much less than the peak's 50 Volt half-width. The issue here is largely one of definition. Because temperature is defined in terms of velocity rather than energy, its value is actually much less than the observed spread in ion energies. In fact, temperature is not a particularly meaningful term in this context. The temperature is deceptive because it implies a much narrower spread in ion energies than is actually the case. The RPA shows that ions have undergone potential drops that differ by 50 Volts, not by 3 volts. Similarly, the data in Figure 4.28 gives the following temperatures

$$T_{ev} = \left(\sqrt{305} - \sqrt{236}\right)^2 = 4.42 \text{ eV} \qquad \text{(high value)}$$

$$T_{ev} = \left(\sqrt{181} - \sqrt{236}\right)^2 = 3.64 \text{ eV} \qquad \text{(low value)}$$

Again, the temperatures differ, though the distribution is much closer to a true Gaussian. The axial ion temperatures vary somewhat, but are generally consistent with Manzella's value of 3.4 eV. The simulated RPA distribution is based on a source axial ion temperature of 34 eV, so it is no surprise that the results fail to agree with the laboratory data. Nevertheless, we use a temperature of 34 eV because it gives better agreement to the observed structure of laboratory measurements of ion current density.

Figures 4.26, 4.27, and 4.29 all show two "peaks" in the ion distribution. The first is a low energy peak near the origin that represents slow moving CEX ions. This peak is too small to be seen in Figure 4.28, but is clearly visible in Figure 4.29 at all angles from the thruster. The second is a high energy peak that represents beam ions coming directly from the anode. This peak is visible in all experimental measurements made at 45 degrees or less. In the simulated data, the low energy peak is narrow and located right at the origin. In Absalamov's data, the low energy peak is slightly wider and is offset from the origin by about 20 V. This may indicate that the RPA itself is floating at some potential relative to the plasma. Another possibilities is that the plume temperature is very high, thus raising the beam potential, but previous measurements of the plume's electron temperature (shown in Figure 2.4) seem to contradict this explanation. The low energy peaks are not offset in the Michigan data or in the simulated results. The high energy peaks are offset from each other by up to 75 eV. Absalamov's peak is at about 275 eV, the peak in the simulated data is just over 200 eV, and the peak in the Michigan data is at about 230 eV. The placement of these peaks is significant because Absalamov's ion energy of 275 eV corresponds to an I_{sp} of 2048 sec, which is considerably higher than experimentally observed I_{sp} of 1735 sec. (excluding cathode flow). Subtracting 25 eV from the peak gives an I_{sp} of 1971 sec., which is still much higher than the experimentally observed value. Michigan's ion energy of 230 eV corresponds to an Isp of 1872 sec., which is within 8% of the measured value. This difference is relatively small and could be due to the spreading of the ion beam after it leaves the thruster. These disagreements suggest that Absalamov's RPA may have been substantially biased with respect to the ambient plasma. Further work is needed to determine the source of this inconsistencies. The Michigan RPA results have been corrected for the local plasma potential and are generally consistent with the measured specific impulse.

The simulated and experimental results differ qualitatively as well. The high energy peak is much broader in the simulated data and, more importantly, the location of the peak varies with the RPA's angle from the centerline. This behavior is consistent with the earlier observation that particles with low energies are more easily turned by the radial electric fields present in the plume (see Section 4.2.1). Both sets of RPA data show no such variation. In fact, although the magnitude of the peak drops rapidly, the location of the peak is the same at all distances from the centerline. Why this occurs is unclear. It may be that the SPT-100 is emitting high energy ions in directions pointing well away from the centerline, though why the thruster would emit high energy ions at angles very far from the centerline is unclear. Such high energy ions could mask the energy shifting effect caused by the plume expansion. An interesting characteristic of the RPA data is the presence of a high energy "tail" of ions with energies hundreds of volts above the 300 V cathode-anode drop. Tails of this type are sometimes produced by plasma instabilities, so the presence of a tail may indicate that instabilities play an important role in the creation of high energy ions moving at large angles relative to the centerline. This tail is also present in the simulated data and is a consequence of the high axial ion temperature assumption. Since the axial ion distribution is relatively wide, the top end of the distribution ends up having velocities well above the mean drift velocity. The tail is therefore a result of the source model and the simulation itself offers no insight into the mechanism by which the source might create such high energy ions. More work is needed to understand the reasons for these disagreements and the role instabilities play in the SPT-100 thruster.

4.2.5 Summary

In summary, the axisymmetric PIC-DSMC model has been compared to multiple sources of experimental data. Overall, the model shows good agreement at the macroscopic level, but not so good agreement at the microscopic level. The model agrees well with measurements of the ion current but compares much less well to RPA data. The reason for these disagreements is unclear, and some inconsistencies are present in the experimental data. We conclude that the model is able to predict large scale phenomena like the ion current density but is not able to reproduce microscopic data like ion distribution functions. In addition, the surface interaction models described in Chapter 3 have been used to calculate erosion rates on Silver and Quartz glass surfaces. These rates have been compared to measured erosion rates for silver interconnectors and solar cell coverglasses. The model shows very good agreement with silver, but is less accurate when predicting the erosion of solar cell coverglasses. More accurate measurements of the coverglass sputtering coefficient are needed to improve these results. The results shown in Figure 4.17 also demonstrate that the ambient neutrals present in ground based tests have a significant effect on the backflow current. This implies that ground based tests will overpredict the degradation which will occur on surfaces on orbiting spacecraft. To study these issues, axisymmetric simulations of a thruster operating in vacuum were also conducted. These results are presented in section 4.4 below.

4.3 Qualitative Analysis

One advantage of the PIC-DSMC plume model is that is can be used to make a more detailed analysis of the plume region than can be done in experimental or analytic studies. This section presents some detailed results of the axisymmetric plume simulation that help illustrate the physics of the plume region. Figure 4.30 shows a phase space plot of the different particles present in the plume simulation. A total of five different types of particles are shown in this plot: beam ions, beam double ions, CEX single ions, CEX double ions, and neutrals. The neutrals are moving at relatively low velocities and form a tight, isotropic distribution of particles around the origin. The CEX ions are clearly visible in this diagram as a wing of low velocity ions which are accelerated sideways and backwards by the electric field in the plume region. The beam ions form a diffuse band of high speed ions generally moving away from the thruster at high speed. Figures 4.31 and 4.32 are snapshots of the simulation showing the location of beam ions and CEX ions at a single point in time. Superimposed on these snapshots are contours of constant potential. Beam ions are defined as all ions and double ions which have not undergone a CEX collision. The lines in the plot indicate the position of particles in the domain and their direction of motion. Figure 4.31 is a snapshot of only the beam ions in the simulation. As one would expect, most of the ions are concentrated along the thruster centerline and move in the axial direction away from the thruster. The ion density and the potential are related by the Boltzmann relationship, so the centerline is also the region of highest electrical potential. The potential falls off with the density, creating radial electric fields that accelerate ions in a radial direction. This electric field causes noticeable spreading in the beam and some ions are visible at high angles with respect to the centerline. Some beam ions also leave the thruster traveling at high angles with respect to the centerline as governed by equation (3.3-5).

Figure 4.32 is a snapshot of only the CEX ions in the simulation. This snapshot corresponds to a simulation of a thruster operating in vacuum, as described in section 4.4 below. It is clear in Figure 4.32 that most of the CEX ions are created in a relatively small region, perhaps two thruster widths wide, in front of the thruster exit. Because the CEX ions have little initial velocity, they are heavily influenced by local electric fields and move largely in the radial direction. These ions clearly form the bulk of the backflow current. It is interesting to note that the CEX ion distribution does form a wing like structure to the side and rear of the thruster. This wing structure has also been seen in ion thruster data and simulations [*Samanta Roy* 1995]. When the beam and CEX ions are plotted together, this wing structure disappears because the beam and CEX ion distributions overlap each other in space. But when the two distributions are plotted separately, the distinctly different dynamics of the CEX ions becomes apparent in the structure of this wing region.

4.4 Axisymmetric Simulations of Thrusters in Space

Ground based tests are carried out in vacuum chambers where the quality of the vacuum is limited by the facility's pumping speed. SPT-100 tests are typically carried out at pressures of 3×10^{-5} Torr, though pressures as low as 6×10^{-6} Torr can be achieved in some facilities. Figure 4.33 shows several plots of the ion current density 60 cm from the exit of an SPT-100 thruster. One line shows the experimentally measured ion current density at a pressure of 2.5×10^{-5} Torr. The next line shows an axisymmetric simulation of a thruster operating at the same ambient pressure. The experimental and simulated results are within a factor of 2-3 across most of the domain. The third set of results is a simulation of a thruster operating in vacuum. These conditions can not be duplicated in ground based tests, but are representative of a thruster operating on a satellite in geostationary orbit. At high angles, the ion current density in vacuum is almost an order of magnitude lower than in simulated ground tests. These results show that the ambient neutrals present in ground based tests can significantly affect experimental results. These neutrals increase the CEX collision rate and lead to experiments that overpredict the backflow which will occur on orbit. Significantly less backflow will be present under operational conditions. Figure 4.34 shows a similar plot comparing simulated results at a pressure of 2.2 x 10^{-6} Torr, which is the lowest pressure achievable in ground based facilities. The simulated results suggest that plume tests conducted at these pressure give a relatively accurate picture of the plume's structure in vacuum. At pressures lower than 2 x 10^{-6} Torr, the neutrals from the thruster come to dominate the background gas and control the CEX collision rate. This observation should give designers more confidence that ground based test in state of the art facilities can replicate the plume's structure under orbital conditions.

The question of most interest to designers is how much damage the plume will cause to satellite surfaces. Figure 4.35 shows the predicted erosion rate for silver and quartz surfaces placed 60 cm from the exit of an SPT-100 operating in vacuum. The results are presented in units of meters per month (720 hours) of operation. The estimated erosion rate varies very strongly with angle from the centerline and varies by roughly an order of magnitude for each 20 degree change in angle. This indicates that a thruster's angle with respect to a surface will have a huge impact on the erosion rate. Another question of interest is how the erosion rate scales with distance from the anode exit. The solid and dotted lines in Figure 4.36 show simulated erosion rates as measured 1 m from the thruster exit. These rates are considerably lower than those shown in Figure 4.35, though silver still erodes 10 microns/month at angles as high as 55 degrees from the centerline. Superimposed on the simulated rates 1 m from the exit are the rates from Figure 4.35 scaled by $1/r^2$. The results show very good agreement, and suggest that it should be possible to scale observed erosion rates using a $1/r^2$ law. The results shown in Figure 4.35 can be combined with this scaling law to produce quick "back of the envelope" estimates of the erosion rate at a given position relative to the thruster. Designers can also use the relationship to scale experimentally measured erosion rates to areas far away from the anode exit.

In summary, axisymmetric simulations of the SPT-100 operating in vacuum suggest that ground based experiments can substantially overpredict the backflow current which will occur under operating conditions. This occurs because the background of neutrals present in ground tests increases the CEX collision rate in the vicinity of the thruster and substantially modifies the experimental results. In addition, axisymmetric simulations of the erosion rate suggest a couple of "rules-of-thumb" which may be useful satellite designers. First, the erosion rate from the thruster falls by about an order of magnitude with each 20 degree increase in the cant angle. Second, the erosion rate scales as the inverse square of the distance from the thruster.

4.5 Scaling of Plume Parameters

The results presented to this point relate directly to the SPT-100 thruster. The PIC-DSMC model is general, however, and could be used to model TAL or ion thrusters as well if appropriate source models were available. This section presents simulations of several imaginary thrusters with exit parameters different from those of a Hall thruster. These simulated results provide insight into the physics of the plume and its sensitivity to different source parameters.

4.5.1 Neutral Flow Rates

The backflow from the plume consists primarily of ions created by charge exchange collisions. The CEX collision rate is determined by the collision cross section and by the neutral density in the plume. We examined the sensitivity of the plume to neutral density by increasing the neutral flow rate while keeping the other conditions in the source model constant. Table 4.3 shows the different conditions simulated in our study.

Neutral Flow Rate (mg./sec.)	Neutral Fraction (η_n) (1-Ionization Fraction)
0.25	0.05
1.25	0.21
2.5	0.34
25	0.84

Table 4.3: Simulated Neutral Flow Rates

In all of the scaling studies, only a single parameter was varied. All other aspects of the source model were kept exactly the same. This is a theoretical, not a physical, assumption. In practice, increasing the neutral flow rate would certainly change the plasma flow rate and the exit velocity profile. However, the intention of this work is to examine how the plume scales in a general sense, not to model physical thrusters. The flow of ionized propellant was held constant at 4.75 mg/sec. Since the neutral velocity was held constant, the CEX mean free path varies inversely with the neutral flow rate. The neutral fraction is a convenient label to for the four cases simulated. Figures 4.37--4.39 show how the plume's potential structure and CEX ions trajectories change with the neutral flow rate. As one would expect, the number of CEX ions increases dramatically with neutral flow rate. These ions not only increase the backflow current, but also change the overall structure of the plume. The wing structure which is barely visible in Figure 4.37 is very well developed in Figure 4.39 due to the high density of CEX ions. The potential structure does not appear to scale directly with mean free path, though the overall structure changes rather dramatically with the flow rate. Figure 4.40 shows how the ion current density changes with neutral flow rate. Overall, the beam current tends to decrease with increased flow rates while the backflow current tends to increase. The measured backflow current actually decreases between $\eta_n = 0.34$ and $\eta_n = 0.84$ because a significant fraction of the CEX ions flow is occurring at angles greater than 120 degrees from the centerline. The decrease in the beam ion current is quite dramatic however, showing that a significant portion of the flow has been converted into backflow current.

4.5.2 Beam Velocity (Specific Impulse)

Earlier in the chapter, the distribution of the beam ions was shown to have a significant impact on the divergence of the beam. The mean ion velocity was also varied to study the effect this would have on the structure of the plume. Changing the beam velocity is equivalent to changing the thruster's specific impulse while leaving all other aspects of the beam intact. The beam velocities simulated are shown in Table 4.4.

Beam Velocity (m/s)	Specific Impulse (sec.)
17000 m/s	1730
8500 m/s	870
4250 m/s	430
1700 m/s	175

Table 4.4: Simulated Beam Velocities

Again, all other aspects of the source model were held constant. The results represent non-physical thrusters which have never been constructed. The mass flow rate was held constant through these simulations. Figures 4.41 and 4.42 show the plume's potential structure and the distribution of CEX ions at specific impulses of 1730 and 175 seconds respectively. The overall structure of the plume is quite similar in both cases. The CEX cross section has a relatively weak dependence on the approach velocity, so the magnitude and direction of the backflow current is largely unaffected. Figures 4.43 and 4.44 show phase space plots at specific impulses of 1730 and 175 seconds respectively. Note that the two figures use different scales for the velocity. In Figure 4.43, the beam ions form a cloud in velocity space that is separated from the CEX and neutral ions. In Figure 4.44, however, the beam ions have merged with the CEX ions and their velocities have clearly been effected by the plume's electric field. Figure 4.45 shows the variation in the ion current density profile with changes in specific impulse. As the beam ion velocity is reduced, the beam diverges more and more until the majority of the backflow current is composed of fast moving beam ions rather than slow moving CEX ions. These results show that, all factors being even, thrusters with low specific impulses will tend to have relatively divergent beams while those with high specific impulses with have relatively narrow beams. As a result, although the CEX portion of the backflow current remains fixed, the overall backflow current tends to increase with decreasing specific impulse.

4.6 Summary

The axisymmetric PIC-DSMC plume model has been used to simulate an SPT-100 operating on the ground and in a full vacuum. The model has been verified through comparison to experimental data and the following conclusions have been reached.

- The PIC-DSMC model accurately simulates large scale phenomena such as measurements of the ion current density in front of the thruster.
- The PIC-DSMC model is currently unable to accurately model small scale phenomena such as the ion distribution function and match the shape of the ion current density distribution. This disagreement occurs because we use a higher than measured ion temperature in our source model.
- The present source model is adequate for modeling large scale phenomena, but consistently overpredicts the ion current at the centerline. In addition, the axial ion temperature used in the model is higher than measured values and the ion distribution is assumed to be Maxwellian when this is not in fact the case. However, the source model is based on the best near field measurements of the ion current density presently available. Better data or better models of the inside of the thruster are needed to improve the source model.
- There are some questions involving the accuracy of existing experimental data. Manzella's measured beam current is larger than the discharge current. Absalamov's RPA data appears to give I_{sp}'s higher than actually observed in SPT thrusters. Whenever practical, we have corrected for these errors when evaluating the accuracy of the simulation.
- Reported LIF measurements and RPA measurements of the axial ion temperature are fully consistent with each other. The measured temperatures do not match values presently used in the ion source model.
- The surface interaction model accurately models the erosion of silver, but is much less accurate when applied to solar cell coverglasses. The inaccuracies appear to arise from the sputtering coefficient. The overall method is valid, but depends heavily on accurate measurements of the sputtering coefficient.

Overall, comparisons with experimental data partially verify the basic PIC-DSMC method. However, some improvements are desirable. The areas requiring the most improvement appear to the be ion source model and the sputtering coefficients used in the surface interaction model. The models appear to control the RPA ion distribution and the erosion model respectively.

One advantage of the PIC-DSMC method is that it can be used to make detailed studies of the structure of the plume region. The following conclusions in particular can be drawn.

- When a Hall Thruster operates in vacuum, the majority of the CEX ions are created within three thruster widths of the channel exit.

- The CEX ions form a wing structure to the side and rear of the plume. However, the main beam has a relatively high divergence angle. As a result, the CEX wing merges with the main beam and is not visible in measurements of the ion current density.
- Increasing the neutral flow rate tends to increase the backflow current and changes the overall structure of the plume region.
- All things being equal, devices with low specific impulses have relatively divergent beams while those with high specific impulses have relatively narrow beams.

Finally, the PIC-DSMC method has been used to simulate an SPT-100 thruster operating in a true vacuum. This matches conditions experienced by a thruster operating on a GEO spacecraft and can not be duplicated in ground based tests. The following conclusions are reached.

- Ground tests may significantly overpredict the backflow current from SPT thrusters, though tests conducted at very low pressures (~2 x 10^{-6} Torr) may accurately reflect operational conditions.
- The erosion rate drops by approximately an order of magnitude with each 20 degrees from the thruster centerline.
- The erosion rate drops with the square of the distance from the thruster exit.

The results shown in this chapter provide insight into the structure of an expanding plasma plume and demonstrate the utility of the PIC-DSMC simulation. Although further improvements can be made, the level of overall agreement validates the overall PIC-DSMC method.

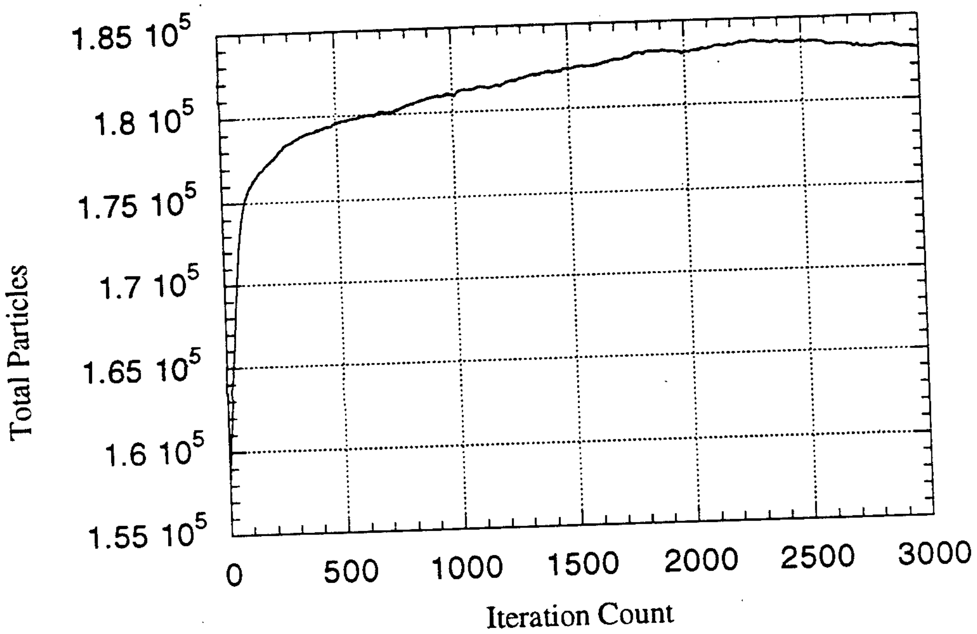


Figure 4.1: Total Particle Count vs. Iteration Time

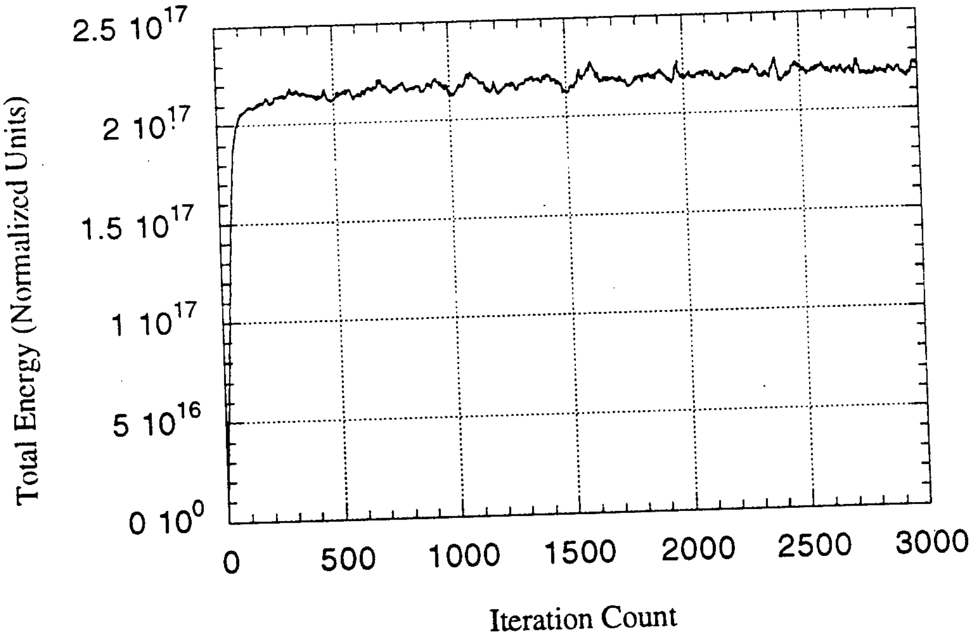


Figure 4.2: Total Energy vs. Iteration Time

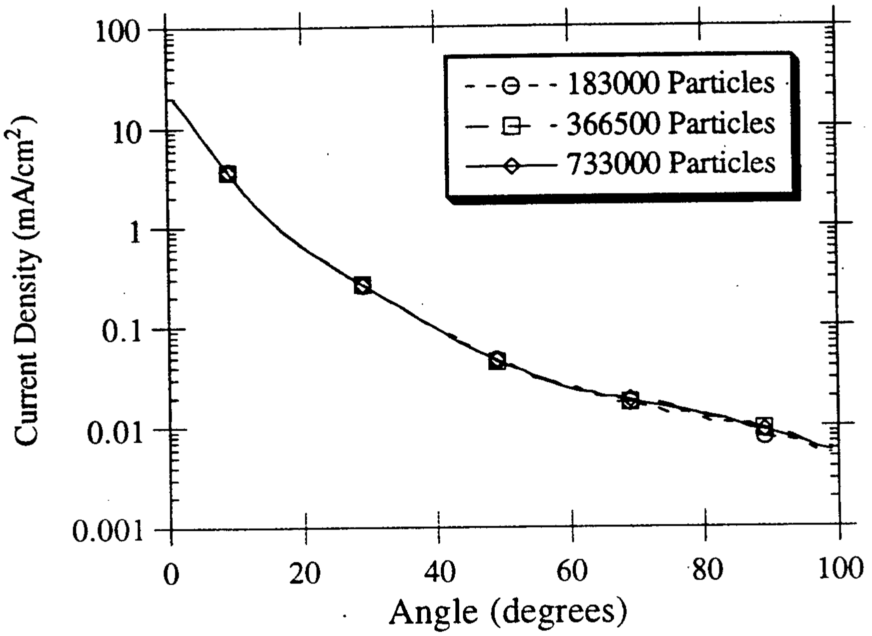


Figure 4.3: Sensitivity to Number of Particles

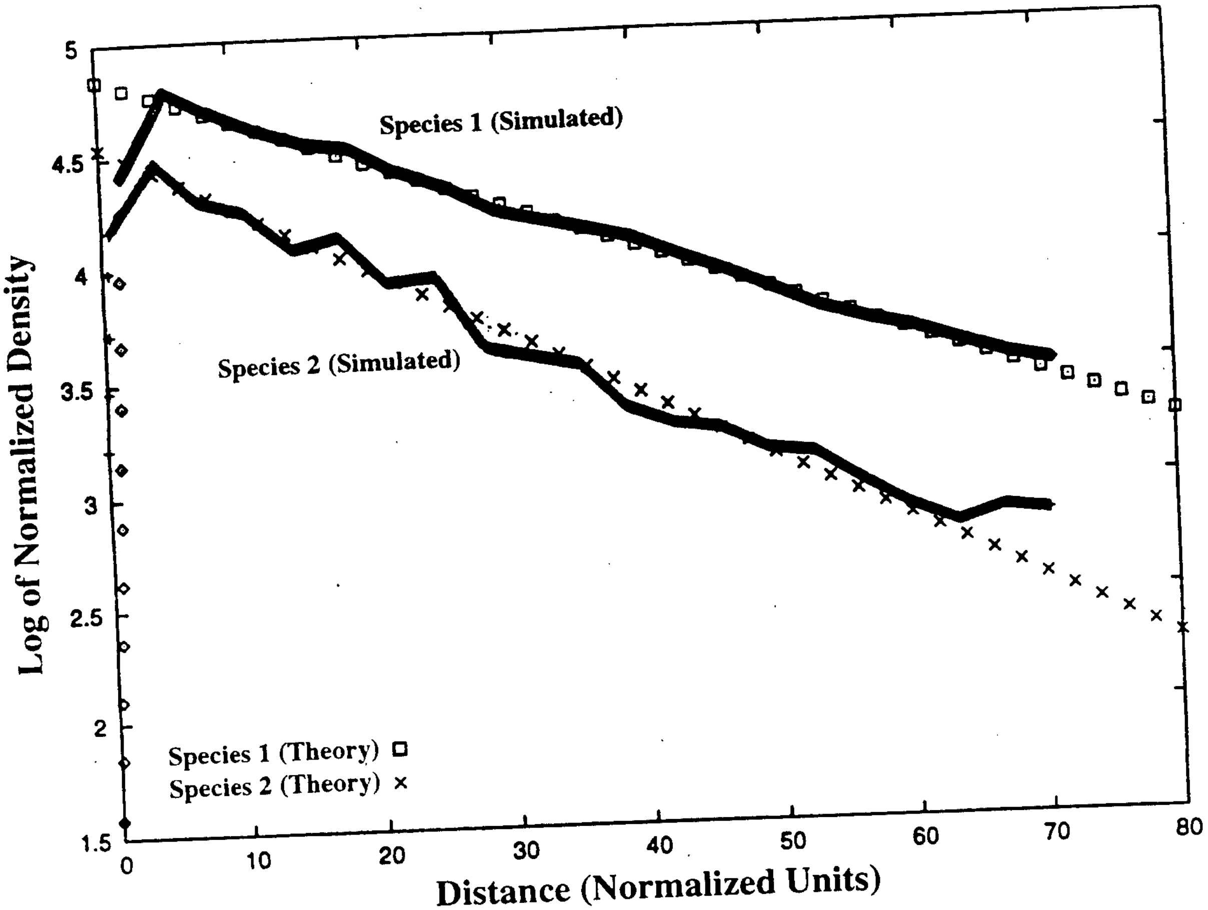


Figure 4.4: Collimated Molecular Beam: Simulation and Theory

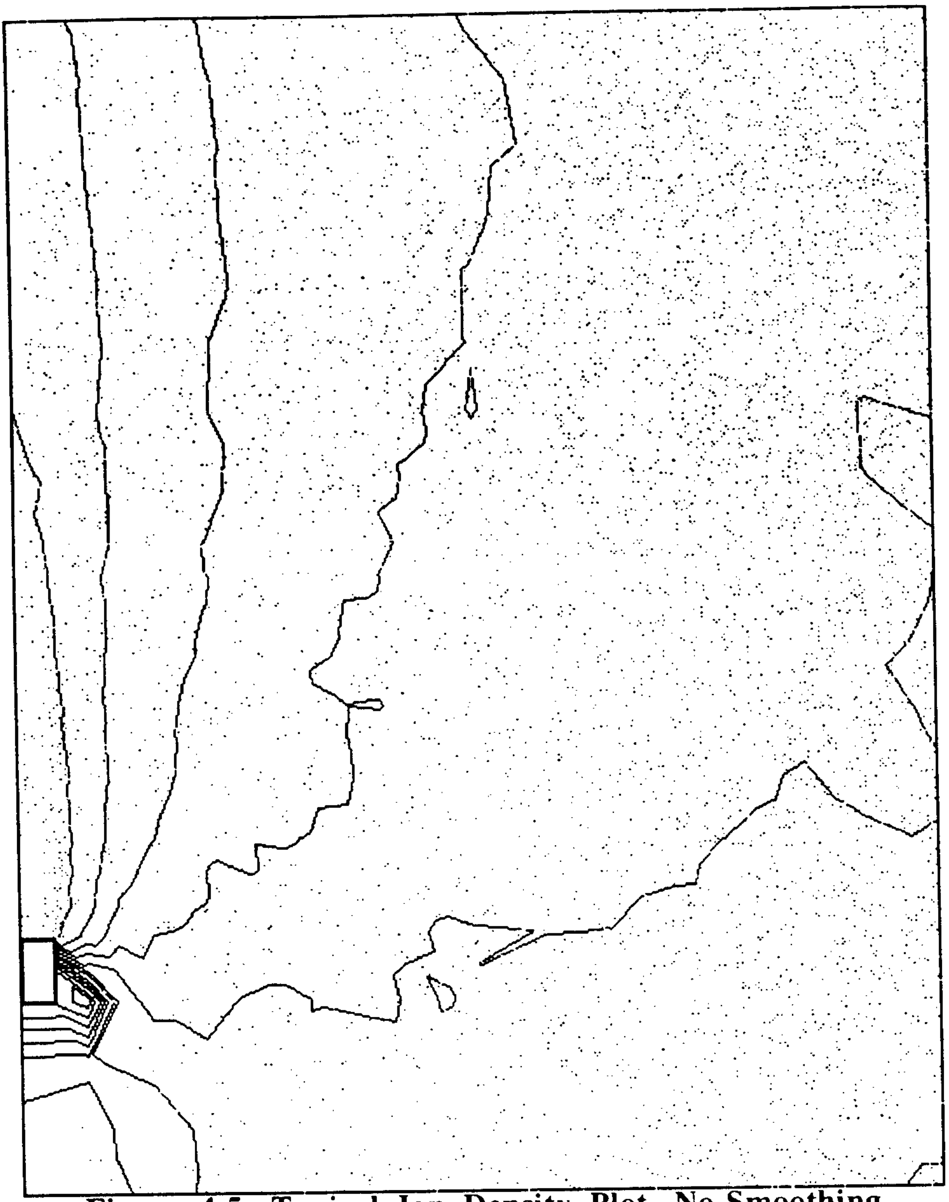


Figure 4.5: Typical Ion Density Plot, No-Smoothing
(Pressure = 5.6 x 10^{-6} Torr)

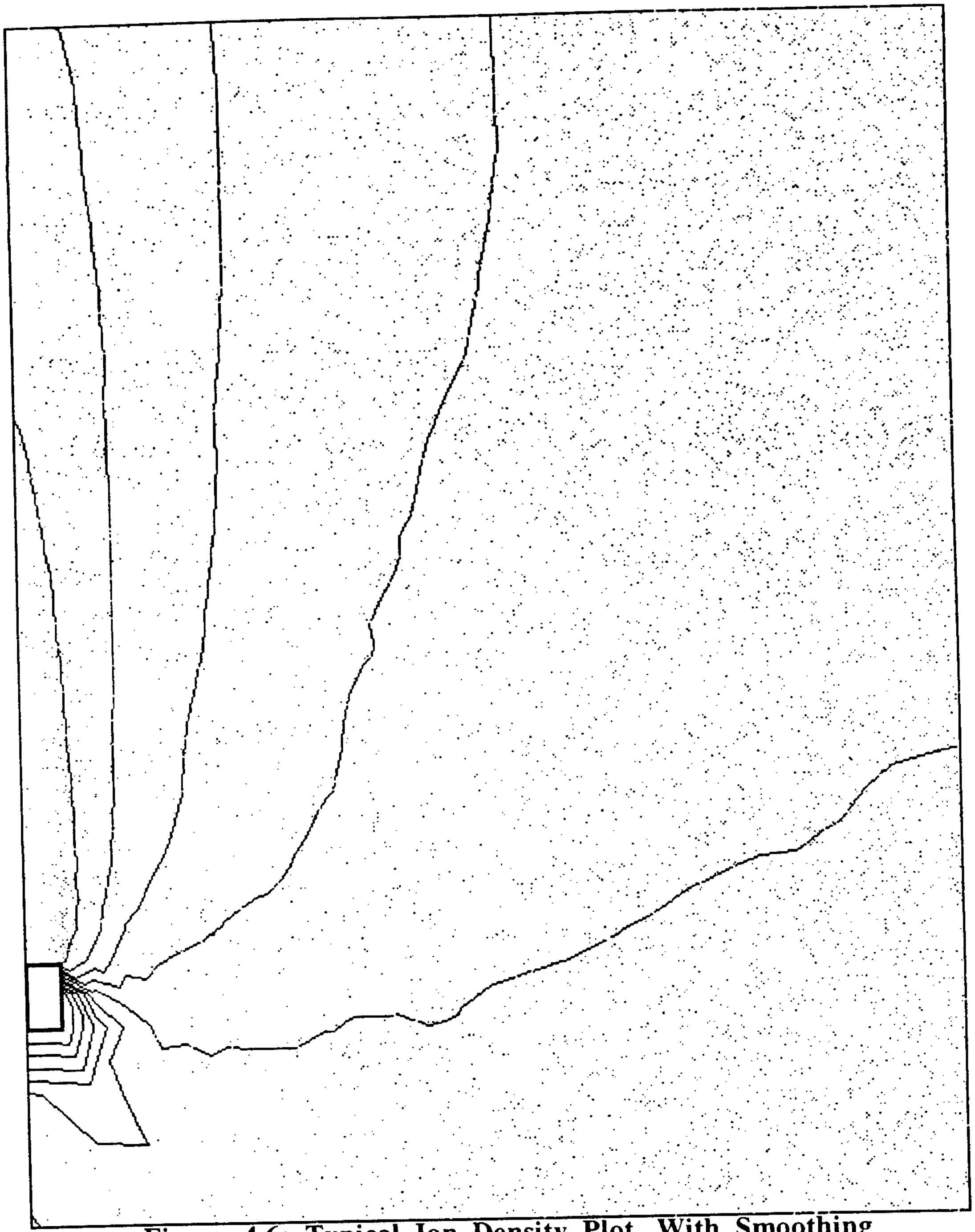


Figure 4.6: Typical Ion Density Plot, With Smoothing
(Pressure = 5.6 x 10^{-6} Torr)

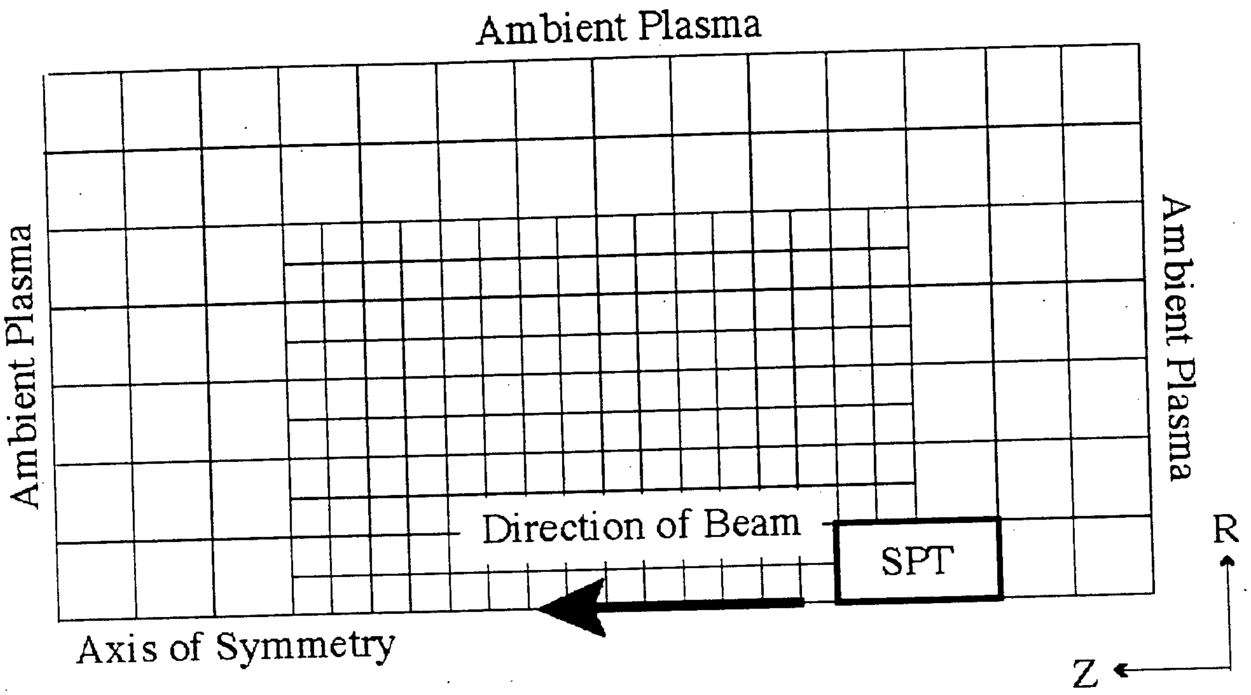


Figure 4.7: Axisymmetric Grid and Domain

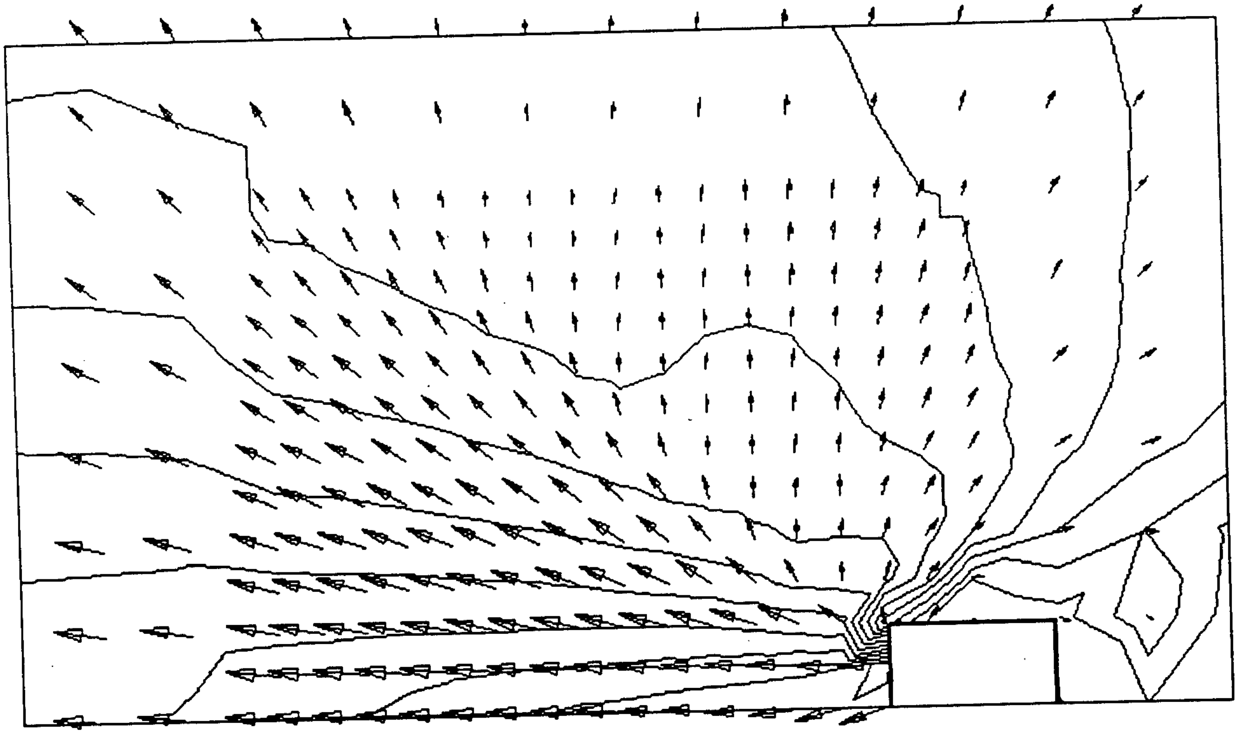


Figure 4.8: Contour Plot of Potential and Vector Plot of Current Density
(Axial Ion Temperature = 3.4 eV)

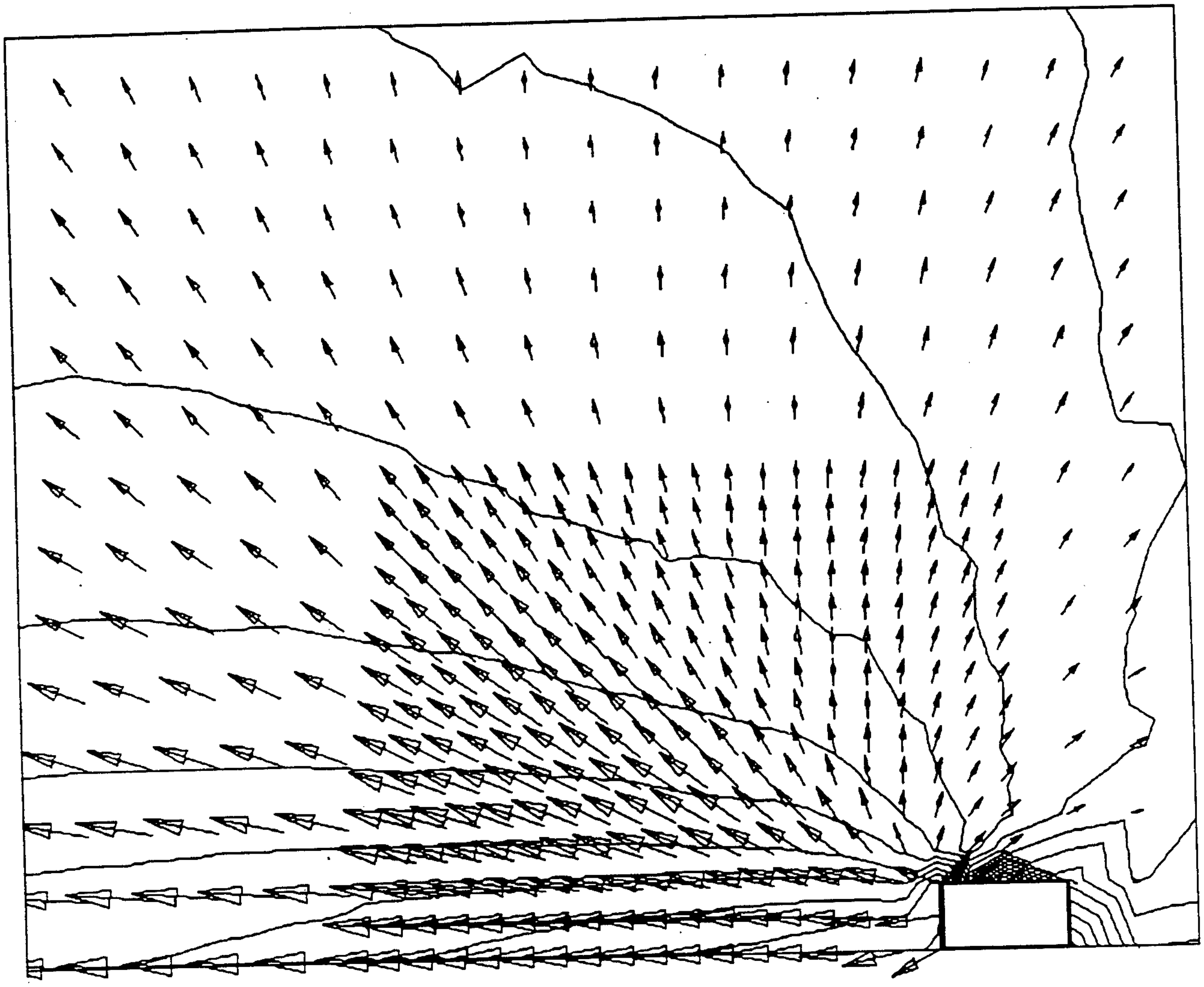


Figure 4.9: Contour Plot of Potential and Vector Plot of Current Density
(Axial Ion Temperature = 34 eV)

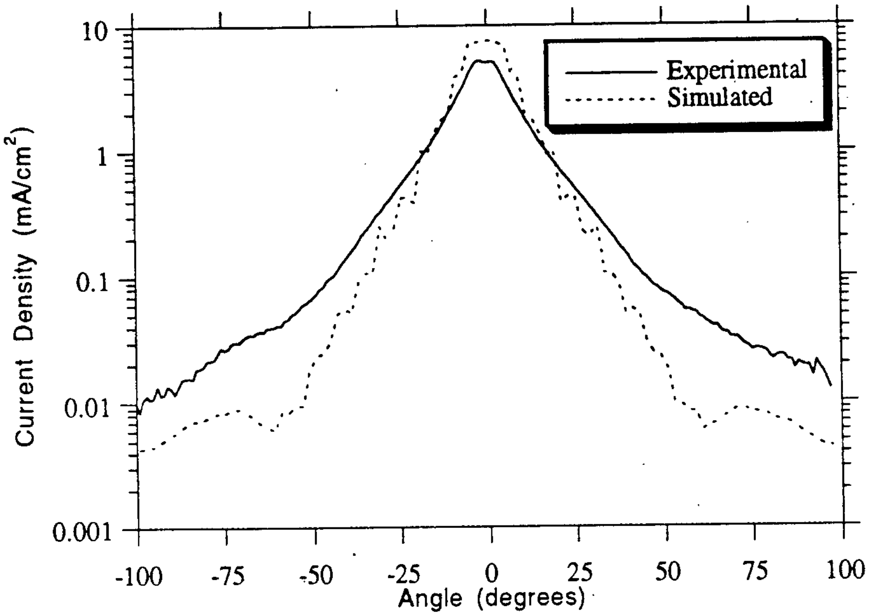


Figure 4.10: Comparison of Ion Current Density Data with Simulated Results
(Axial Ion Temperature = 3.4 eV, r = 60 cm, P = 2.2 x 10^{-6} Torr)

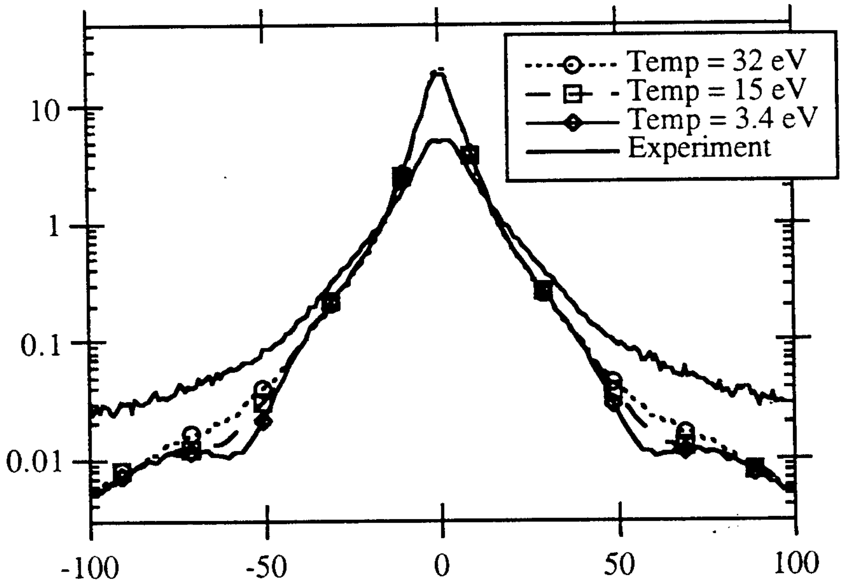


Figure 4.11: Simulated Ion Current at Different Temperatures and Experiment
(r = 60 cm, P = 5.6 x 10^{-6} Torr)

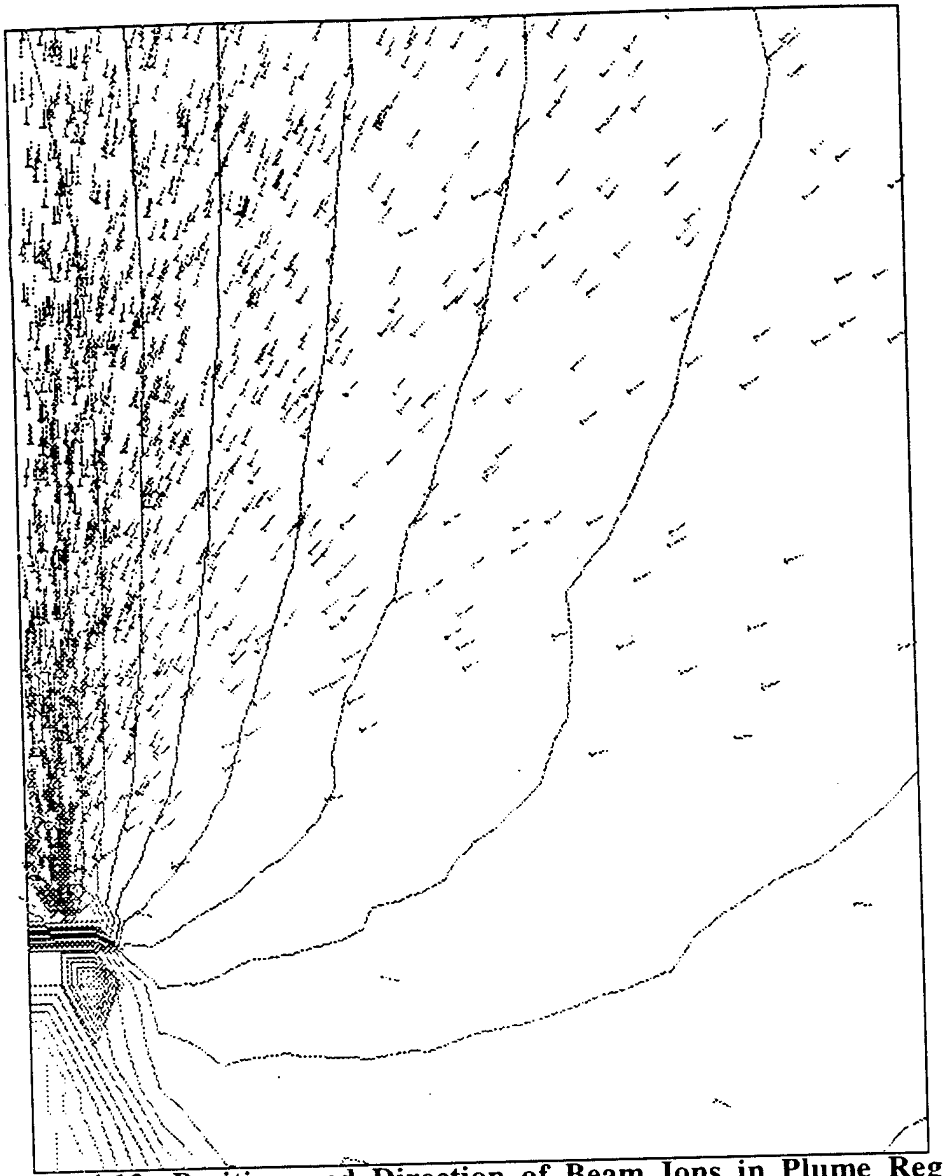


Figure 4.12: Position and Direction of Beam Ions in Plume Region

(Pressure = 2.2 x 10^{-6} Torr)

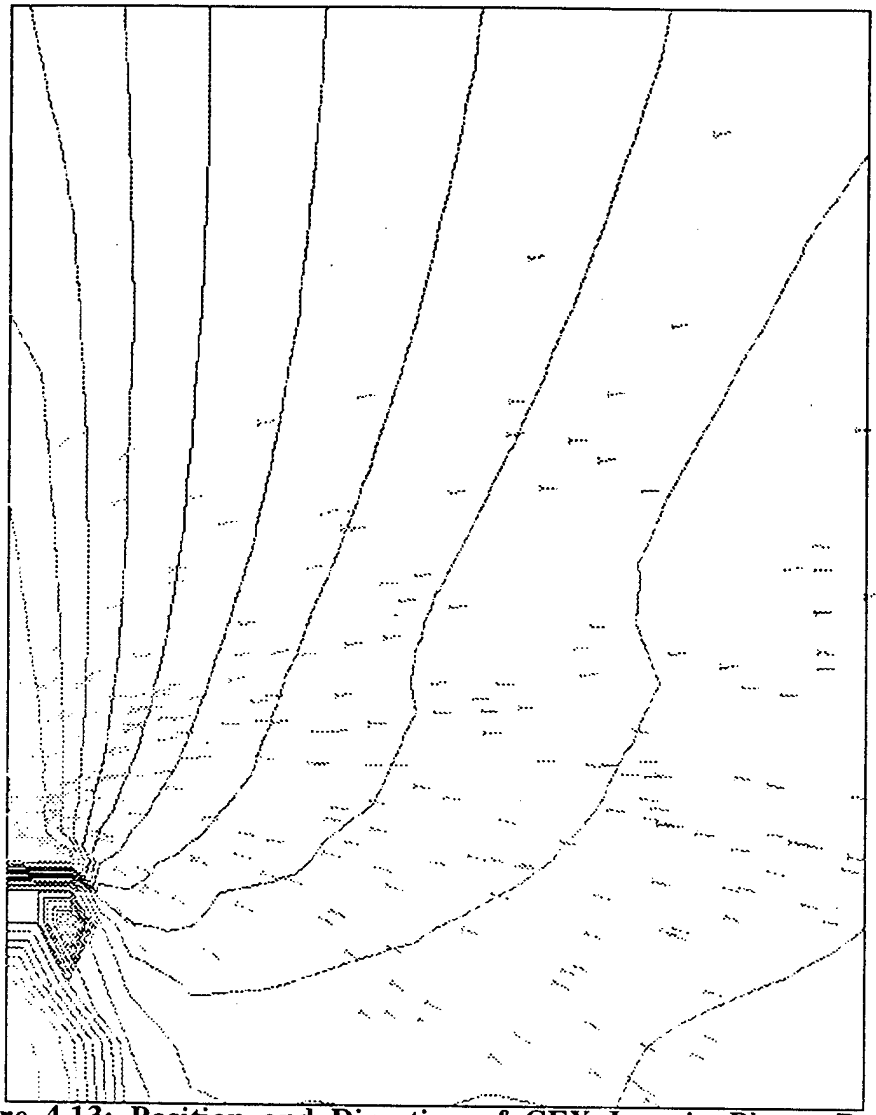


Figure 4.13: Position and Direction of CEX Ions in Plume Region
(Pressure = 0 Torr)

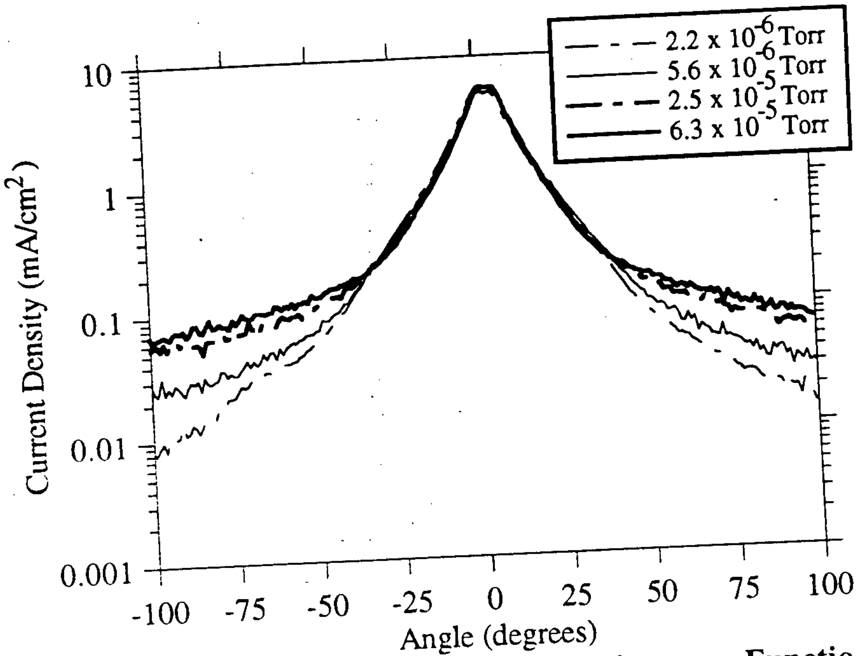


Figure 4.14: Measurements of Ion Current Density as a Function of Facility Pressure (z = 60 cm)

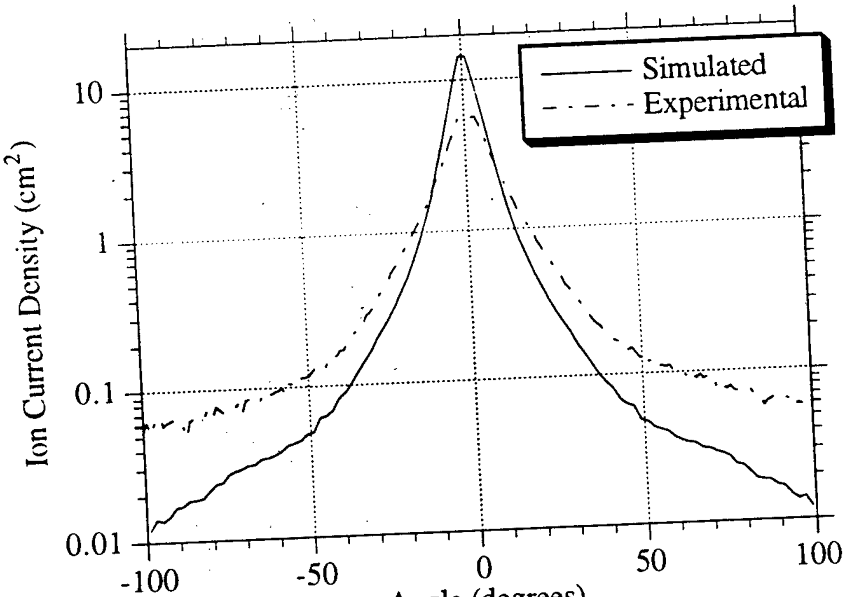


Figure 4.15: Comparison of Ion Current Density with No Compensating Flow
(Pressure = 2.5 x 10^{-5} Torr)

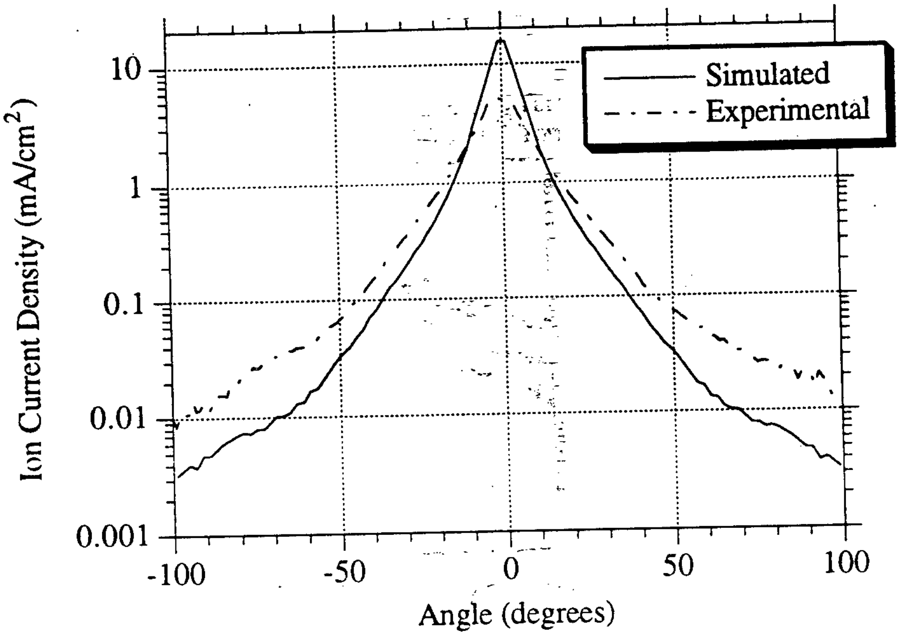


Figure 4.16 Comparison of Ion Current Density with No Compensating Flow
(Pressure = 2.2 x 10^{-6} Torr)

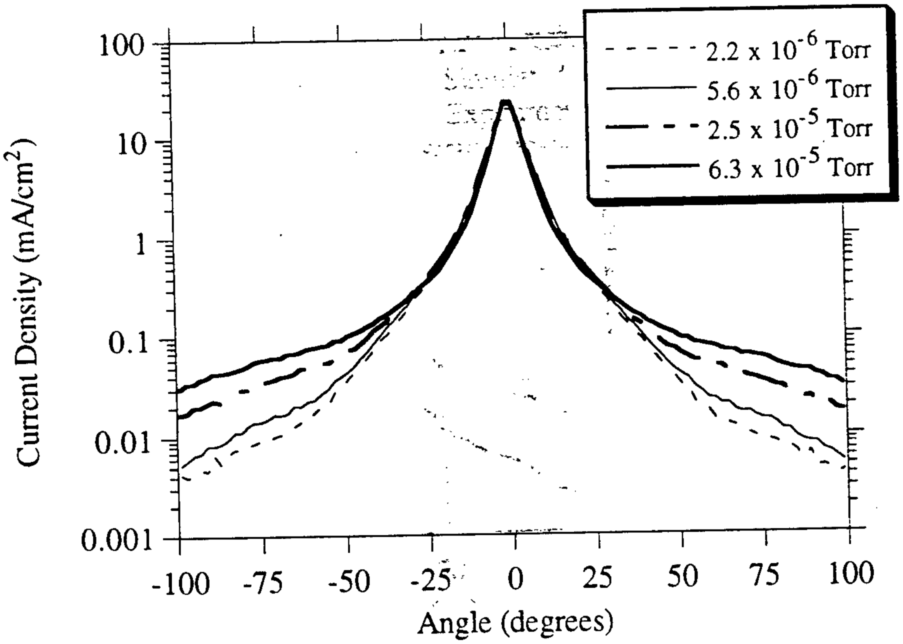


Figure 4.17: Simulated Ion Current Density as a Function of Ambient Pressure
(z = 60 cm)

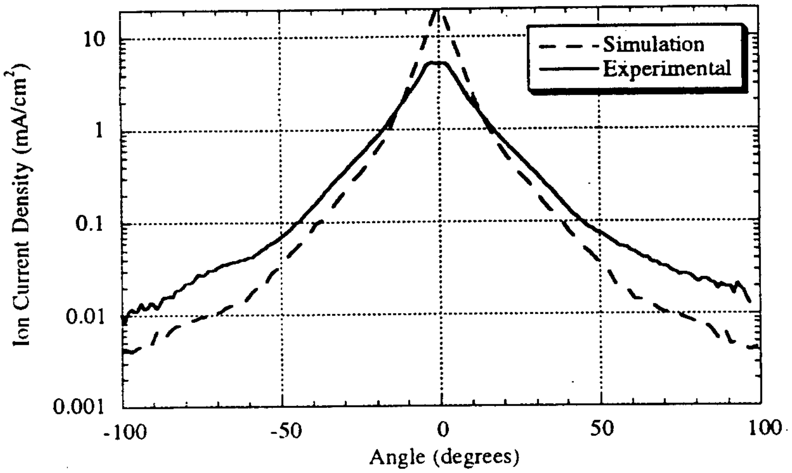


Figure 4.18: Simulated Ion Current Density vs. Data (P= 2.2 x 10^{-6} Torr)

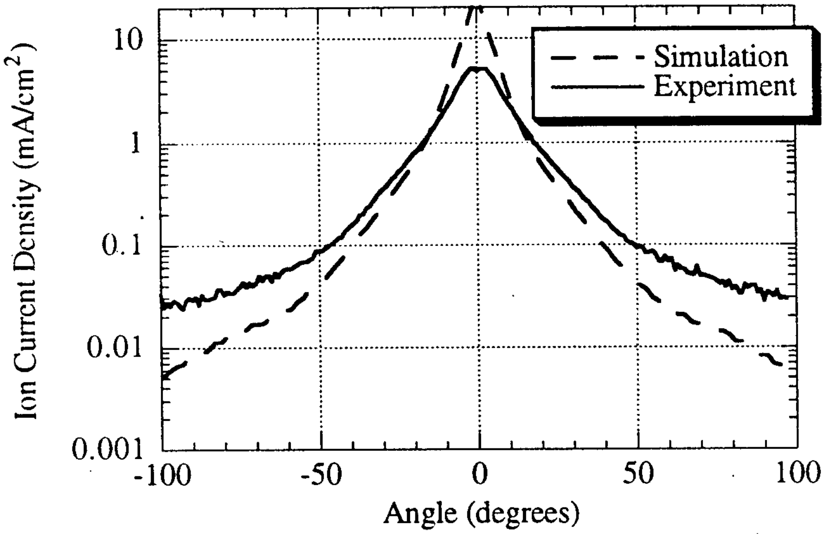


Figure 4.19: Simulated Ion Current Density vs. Data (P = 5.6 x 10^{-6} Torr)

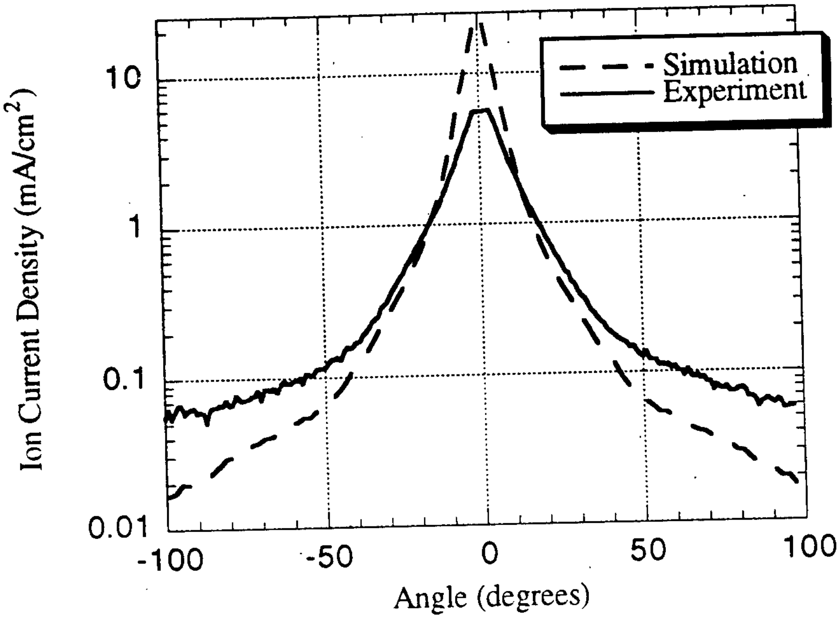


Figure 4.20: Simulated Ion Current Density vs. Data (P = 2.5 x 10^{-5} Torr)

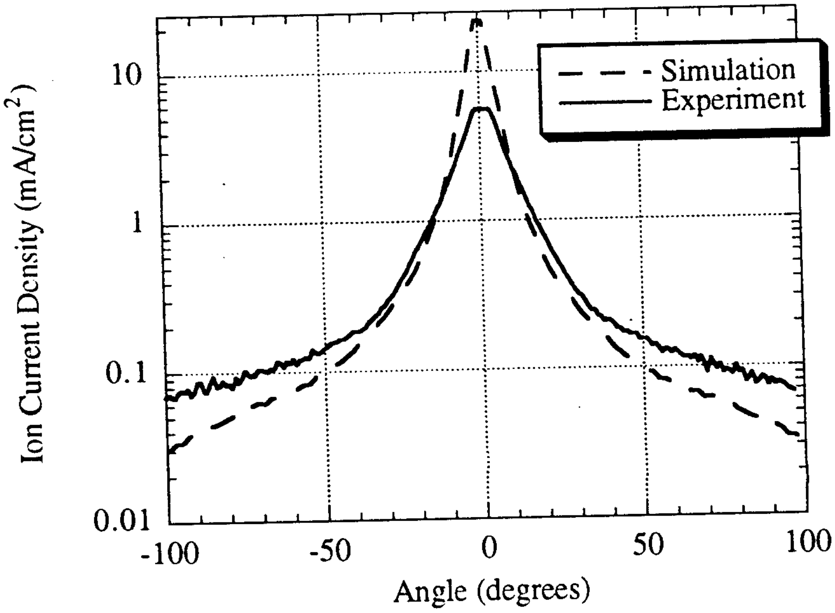


Figure 4.21: Simulated Ion Current Density vs. Data (P = 6.3 x 10^{-5} Torr)

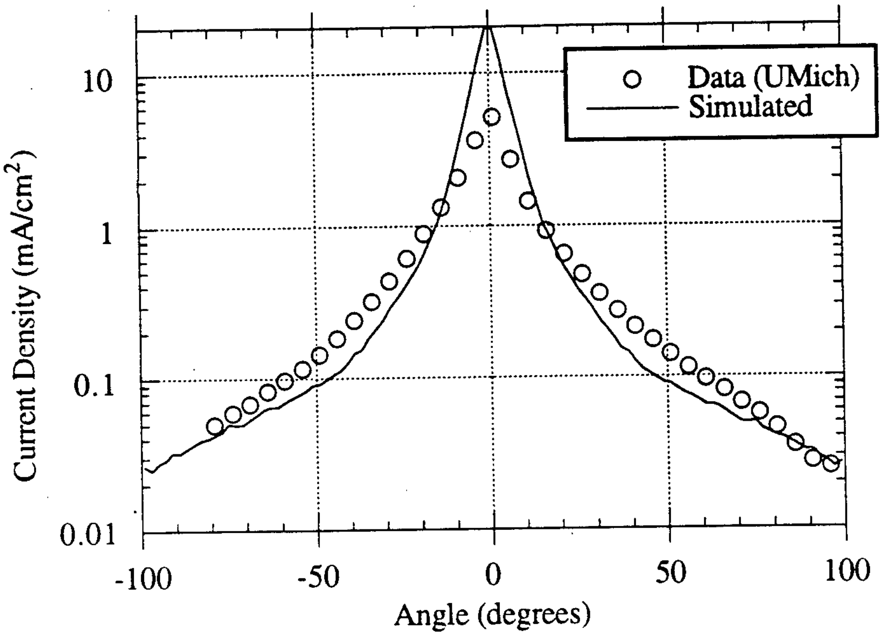


Figure 4.22: Comparison with Data from University of Michigan
(Pressure = 5.2 x 10^{-5} Torr)

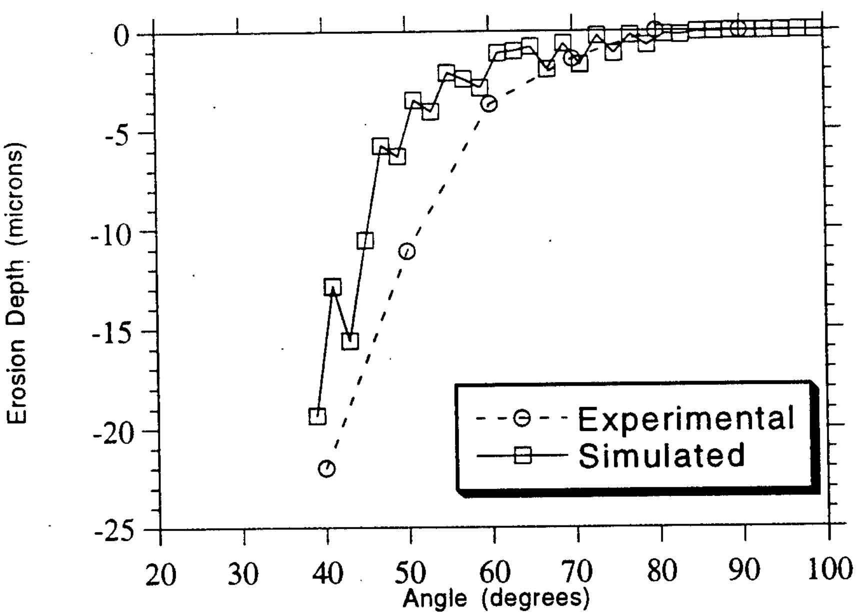


Figure 4.23: Calculated Silver Erosion Rates vs. Experimental Data
(z = 1 m)

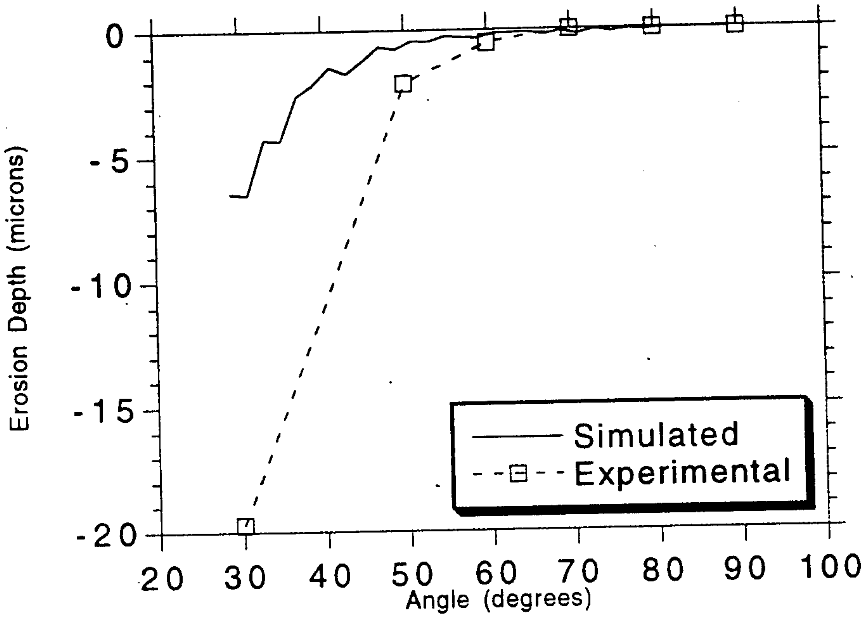


Figure 4.24: Calculated Coverglass Erosion Rates vs. Experimental Data *(z = 1 m)*

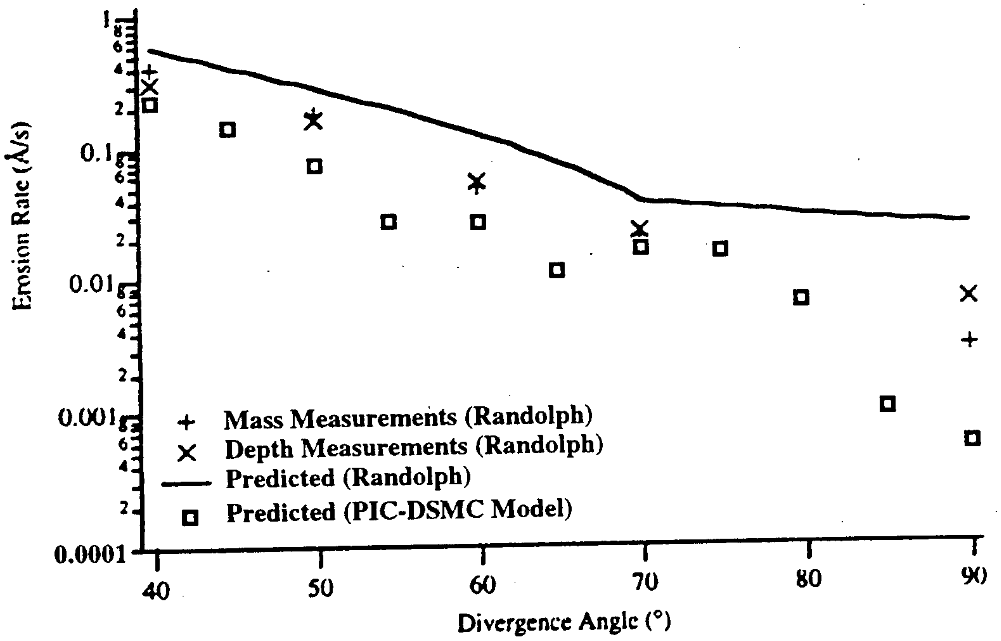


Figure 4.25: Comparison to Silver Sputtering Rates Predicted by Randolph [1994]

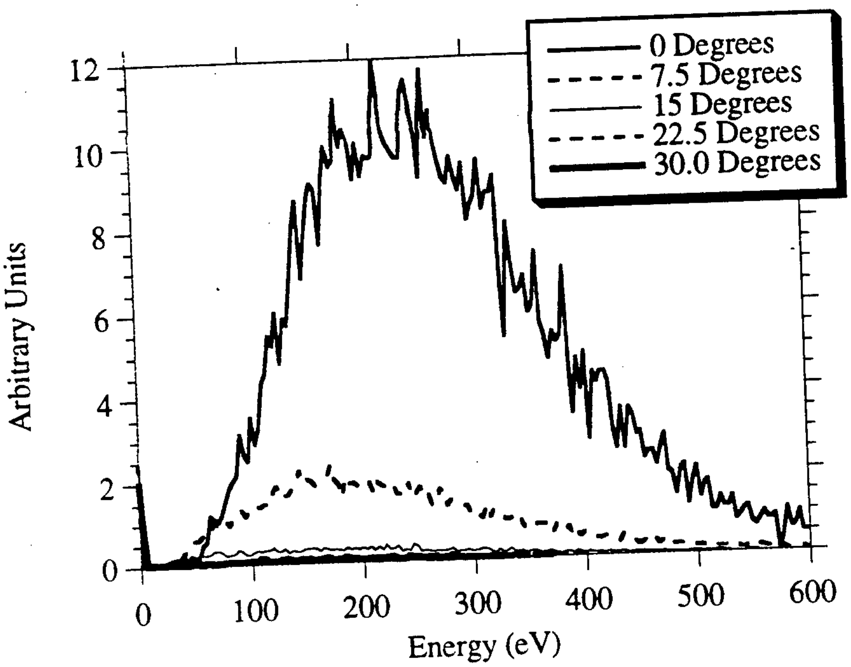


Figure 4.26: Simulated RPA Data Showing Ion Energy Distribution
(z = 60 cm)

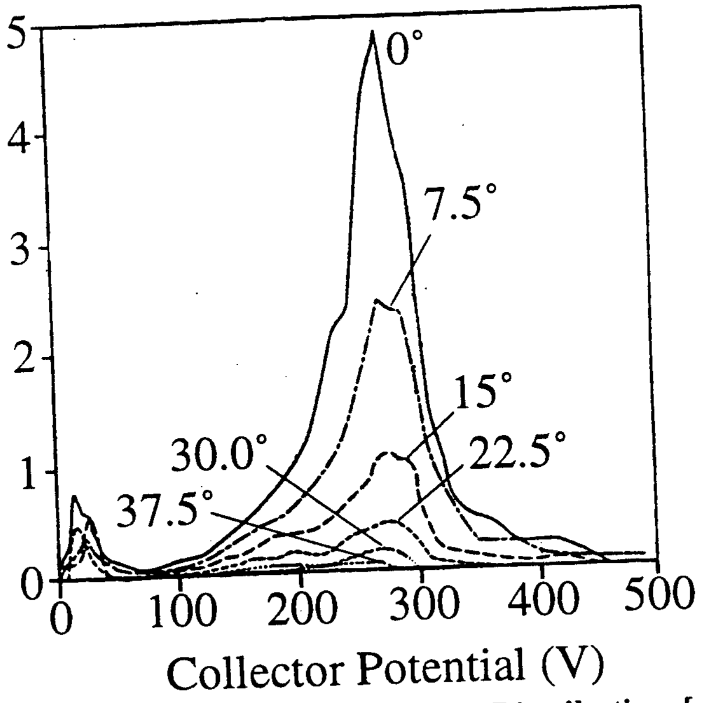


Figure 4.27: RPA Measurements of Ion Energy Distribution [*Absalamov* 1992]
(z = 1 m)

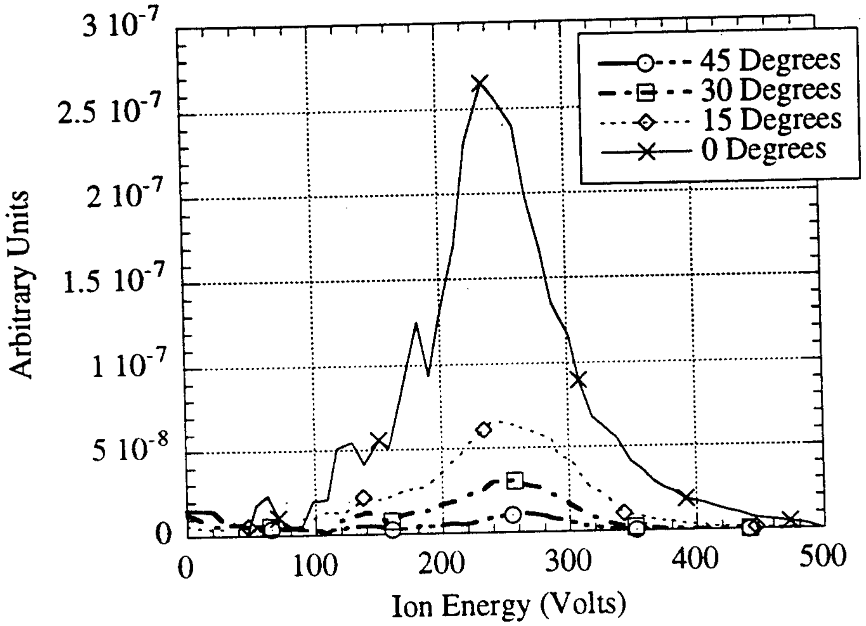


Figure 4.28: RPA Measurements of Ion Energy Distribution [Marrese/Gallimore]
(Pressure = 5.2 x 10^{-5} Torr)

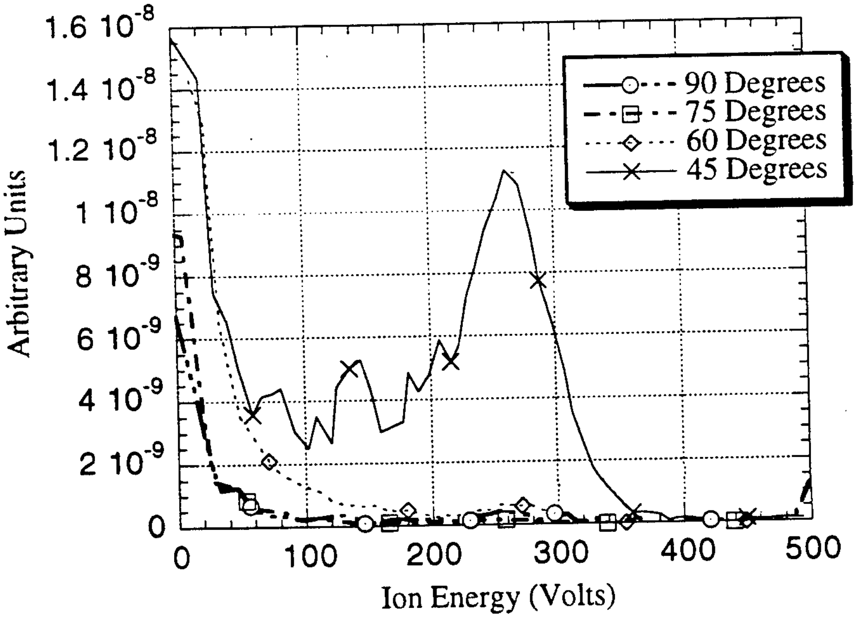


Figure 4.29: RPA Measurements of Ion Energy Distribution [Marrese/Gallimore]
(Pressure = 5.2 x 10^{-5} Torr)

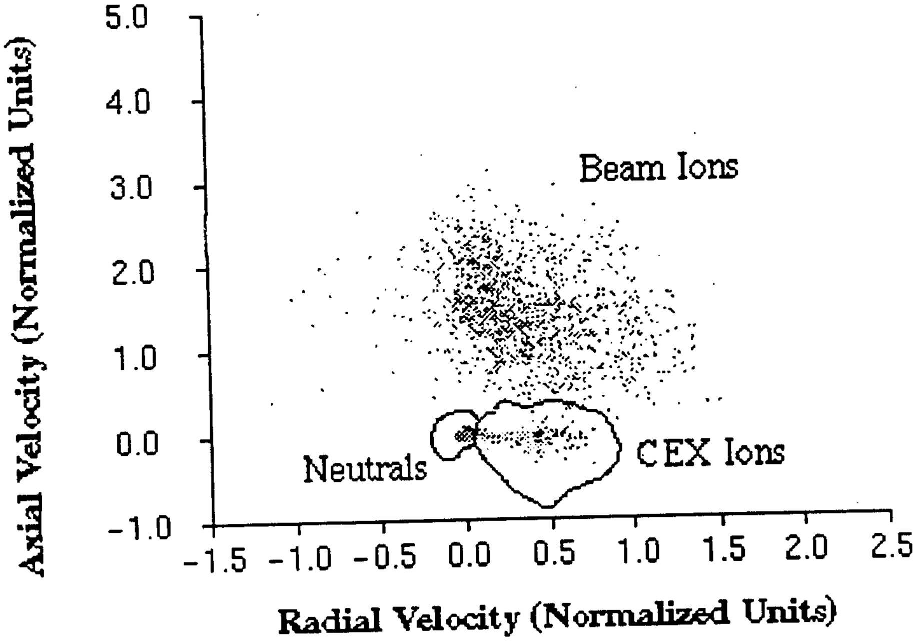


Figure 4.30: Phase Space Plot of Hall Thruster Plume
(Pressure = 2.2 x 10^{-6} Torr)

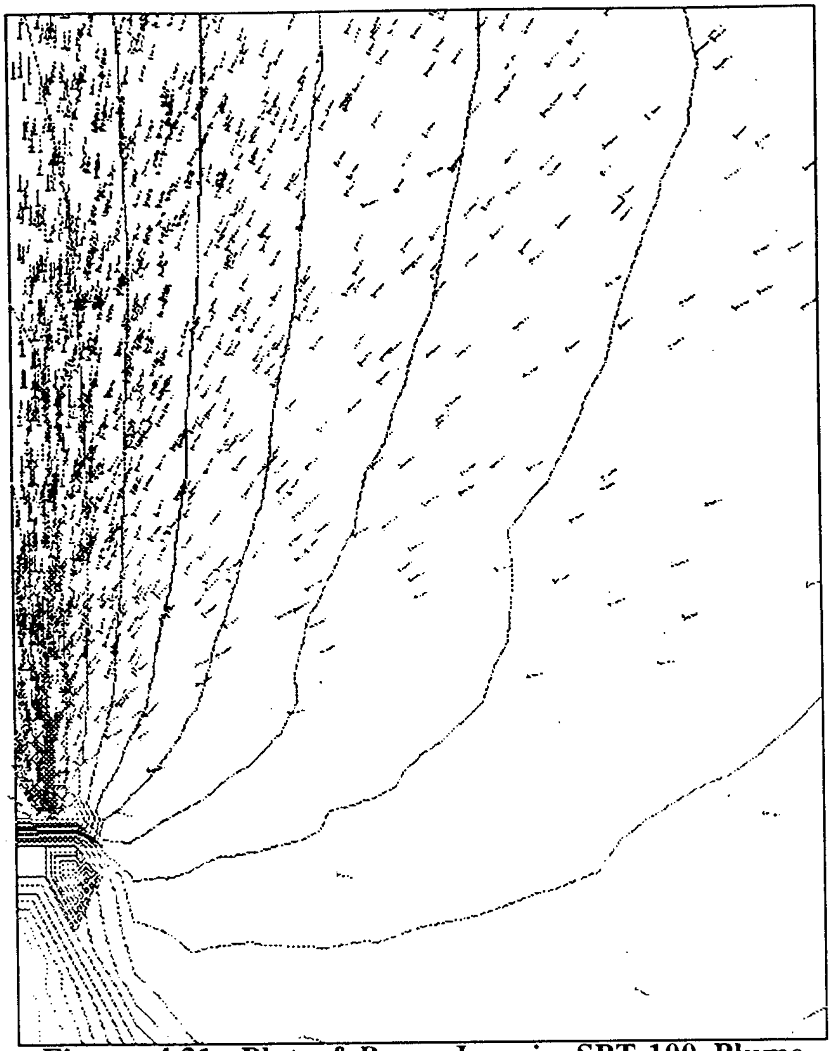


Figure 4.31: Plot of Beam Ions in SPT-100 Plume
(Pressure = 2.2 x 10^{-6} Torr)

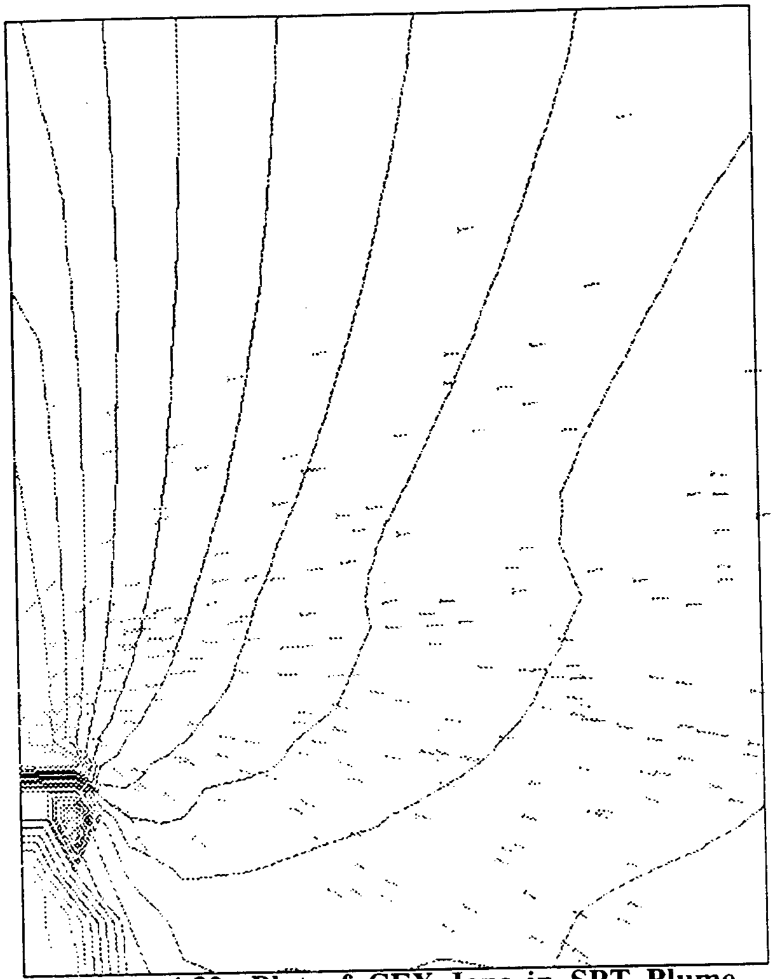


Figure 4.32: Plot of CEX Ions in SPT Plume
(Pressure = 0 Torr)

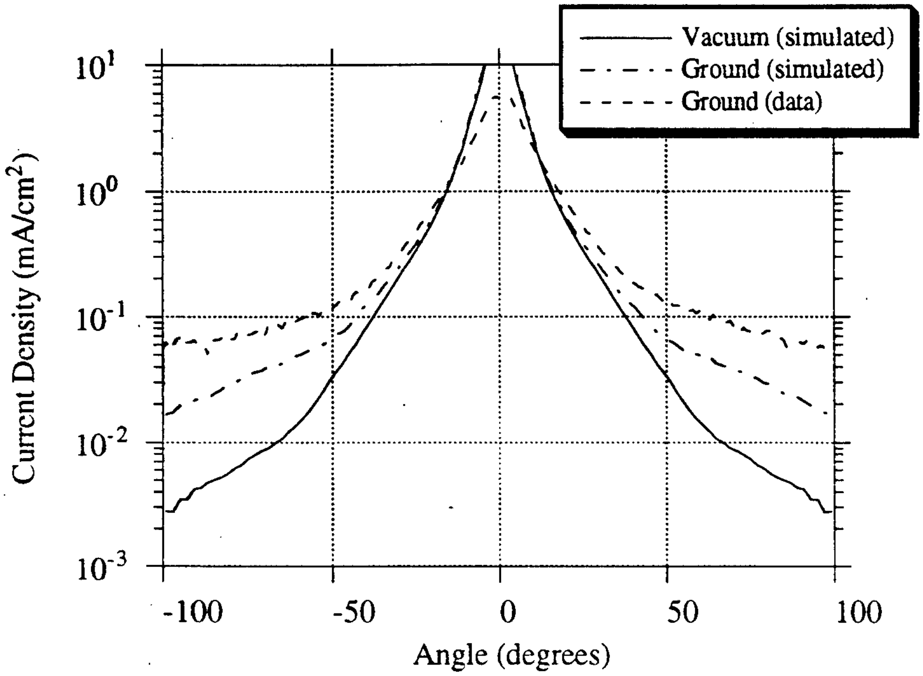


Figure 4.33: Ion Current Density 60 cm from Anode Exit on Ground *(Pressure = 2.5 x 10^{-5} Torr) and in Vacuum*

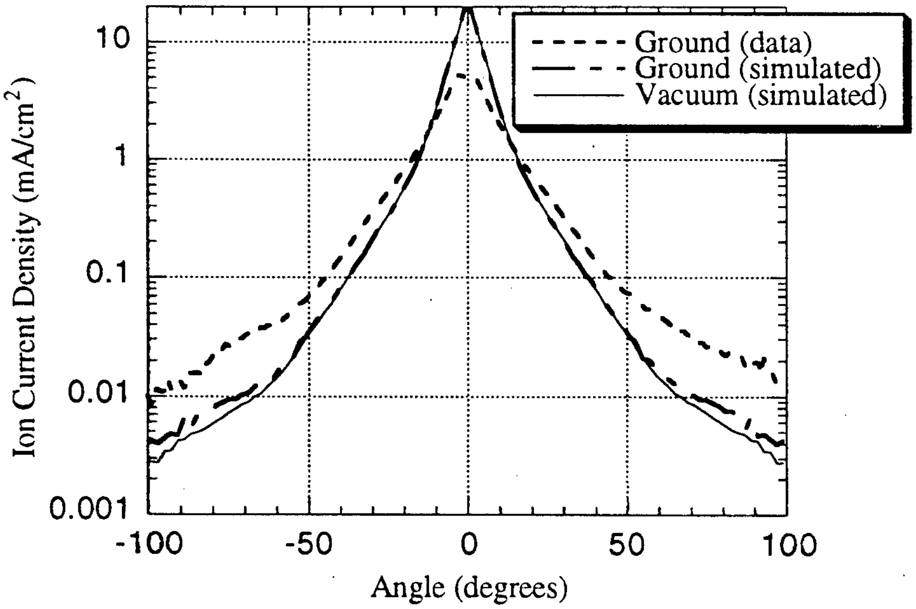


Figure 4.34: Ion Current Density 60 cm from Anode Exit on Ground *(Pressure = 2.2 x 10^{-6} Torr and in Vacuum)*

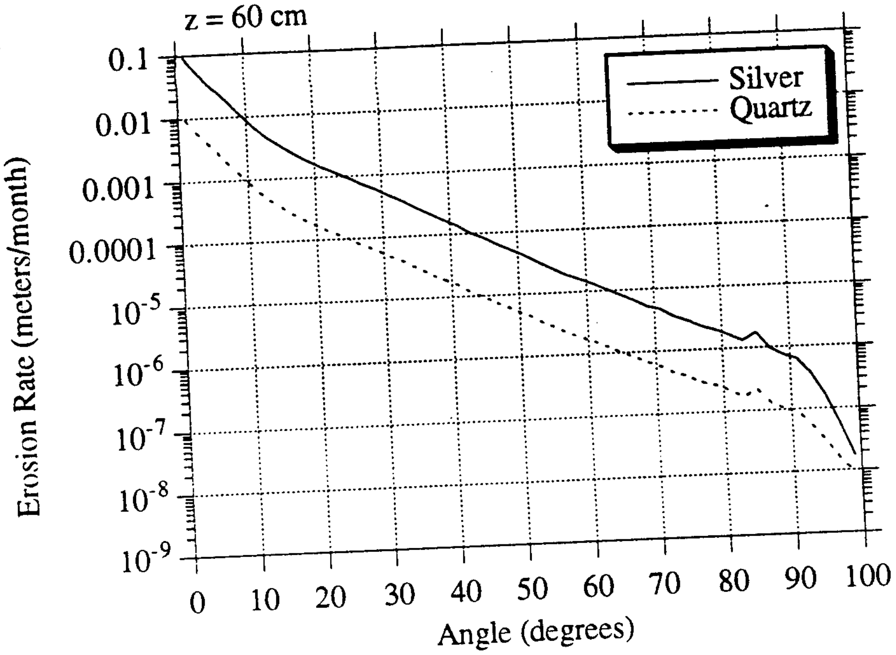


Figure 4.35: Predicted Erosion Rates for Silver and Quartz
(z = 60 cm, Vacuum)

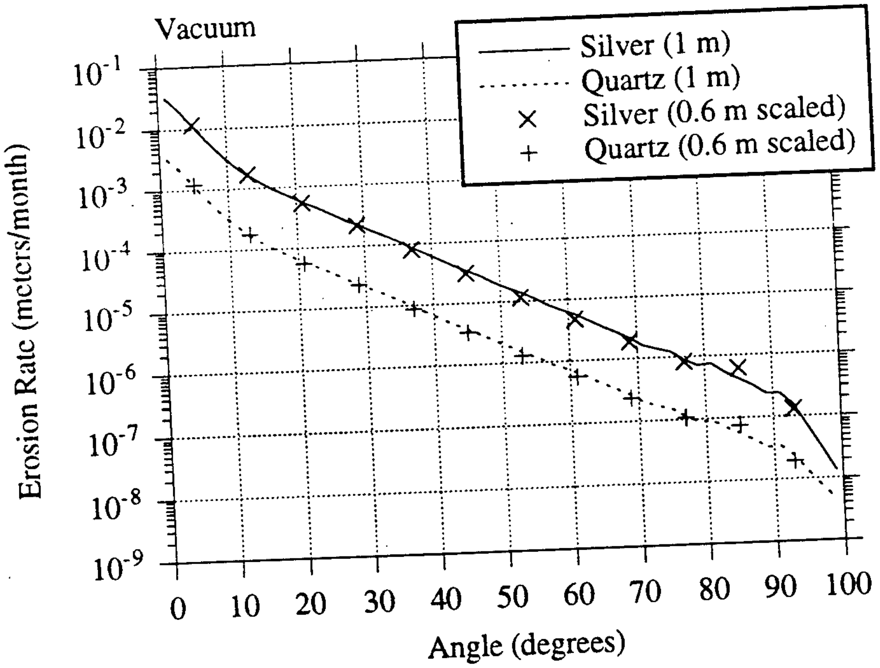


Figure 4.36: Comparison of Measured and Scaled Erosion Rates
(z = 1 m, Vacuum)

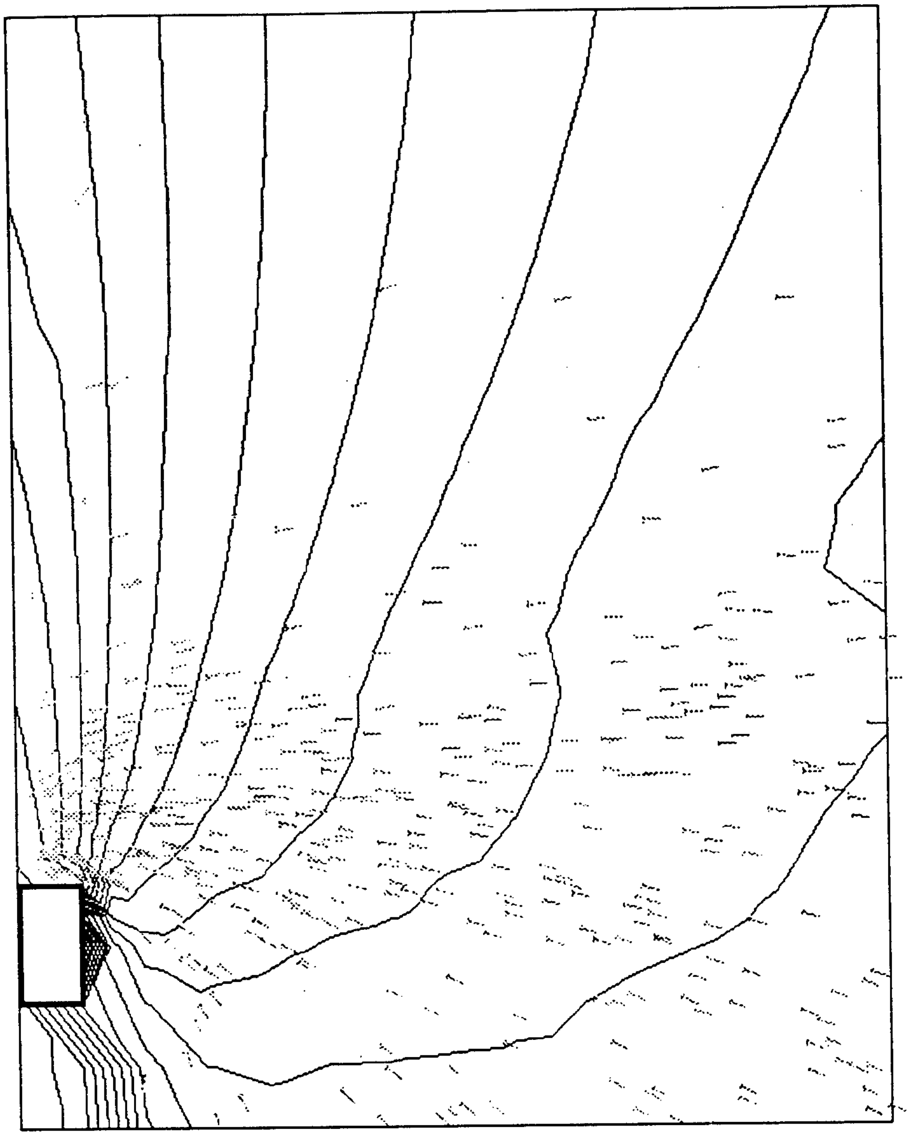


Figure 4.37: CEX Ions and Potential Contours (η_n = 5%)

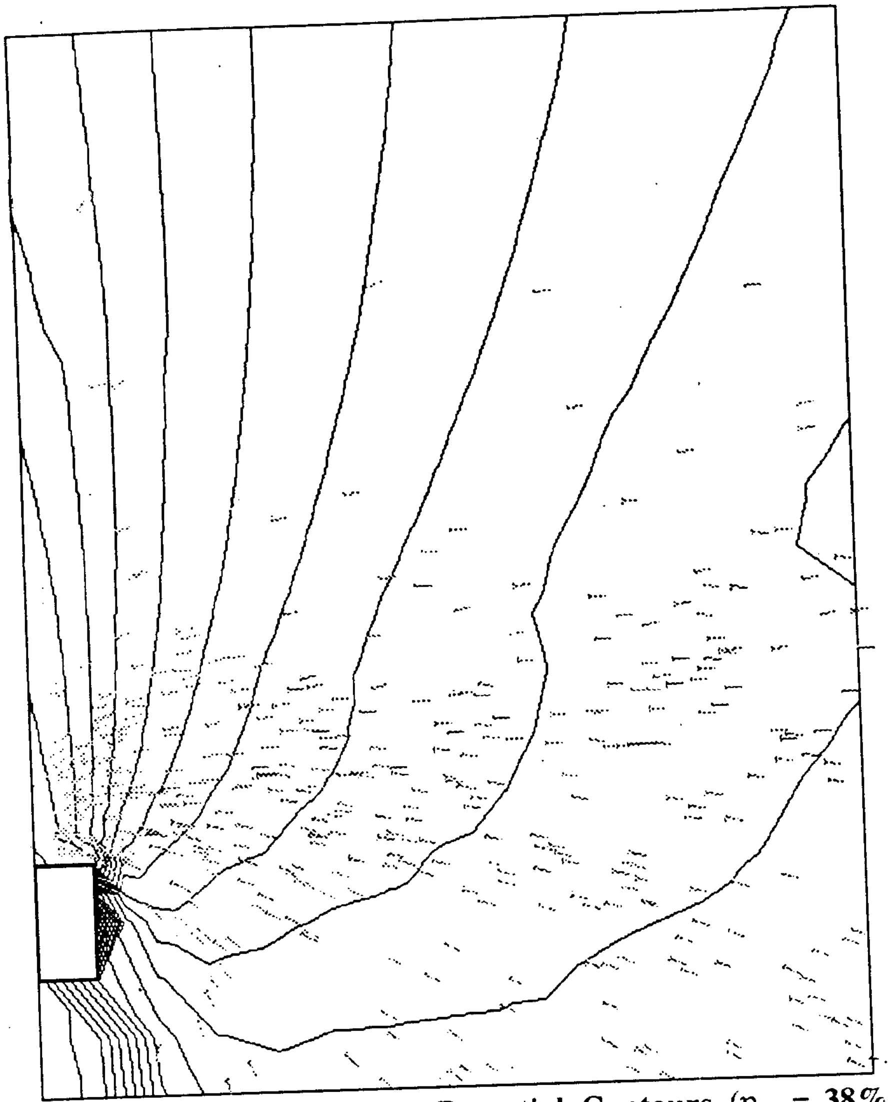


Figure 4.38: CEX Ions and Potential Contours (η_n = 38%)

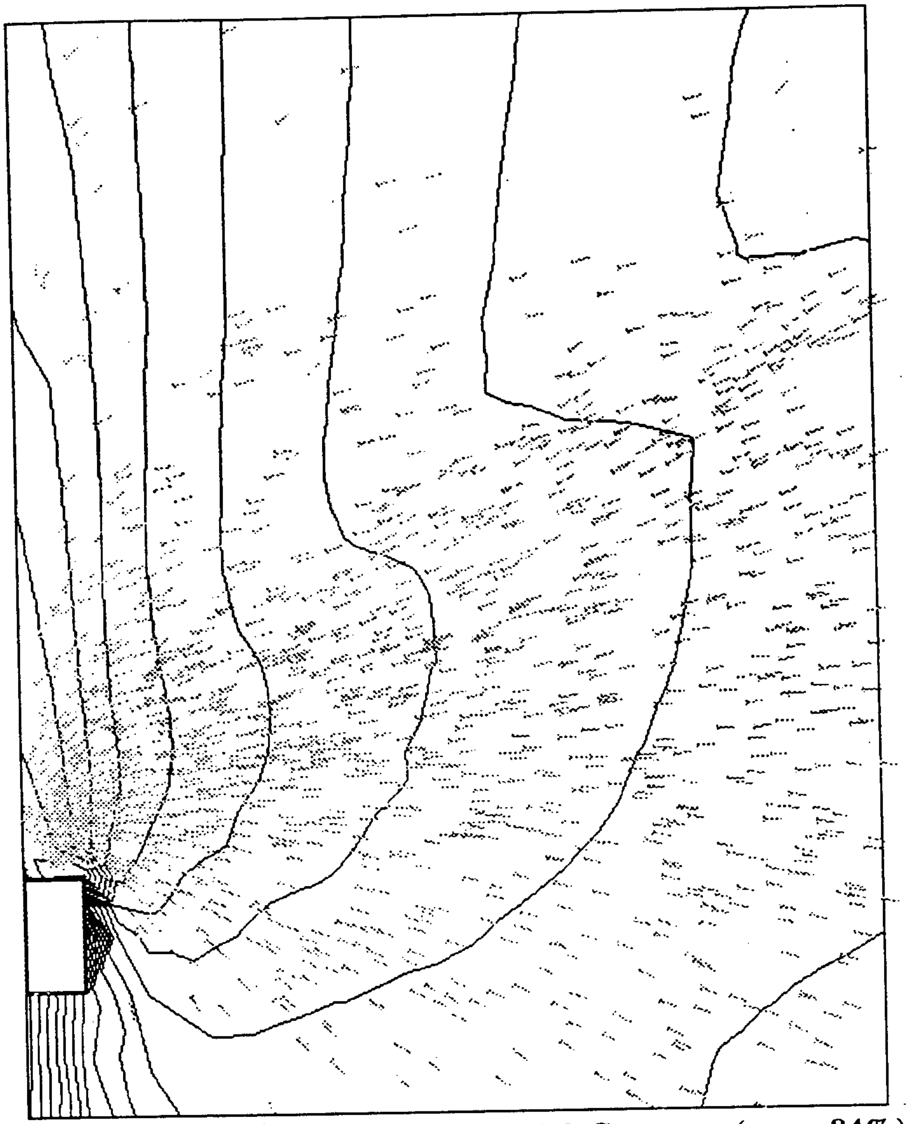


Figure 4.39: CEX Ions and Potential Contours (η_n = 84%)

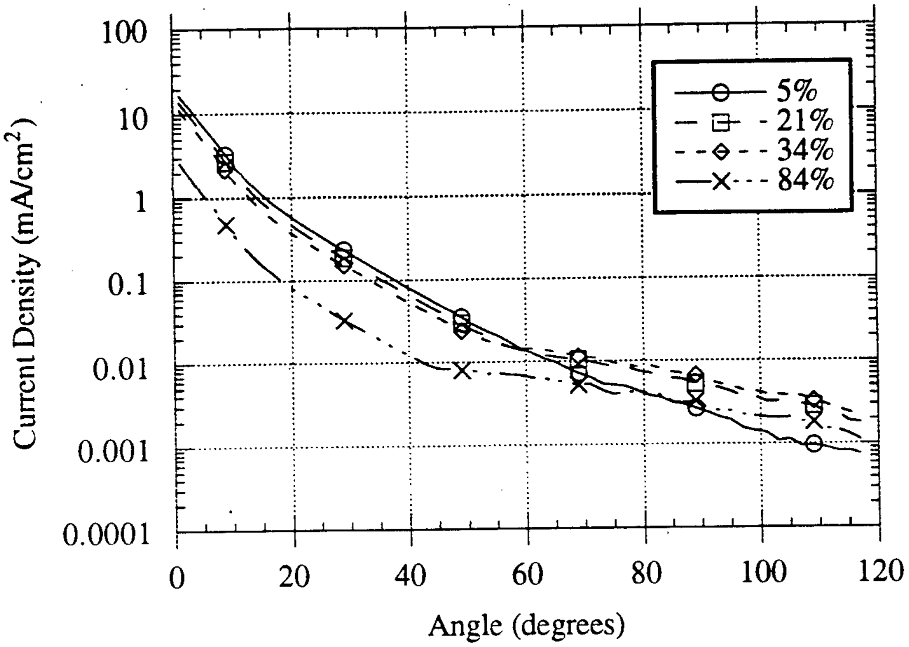


Figure 4.40: Ion Current Density with Different Neutral Flow Rates
(Radius = 0.6 meters)

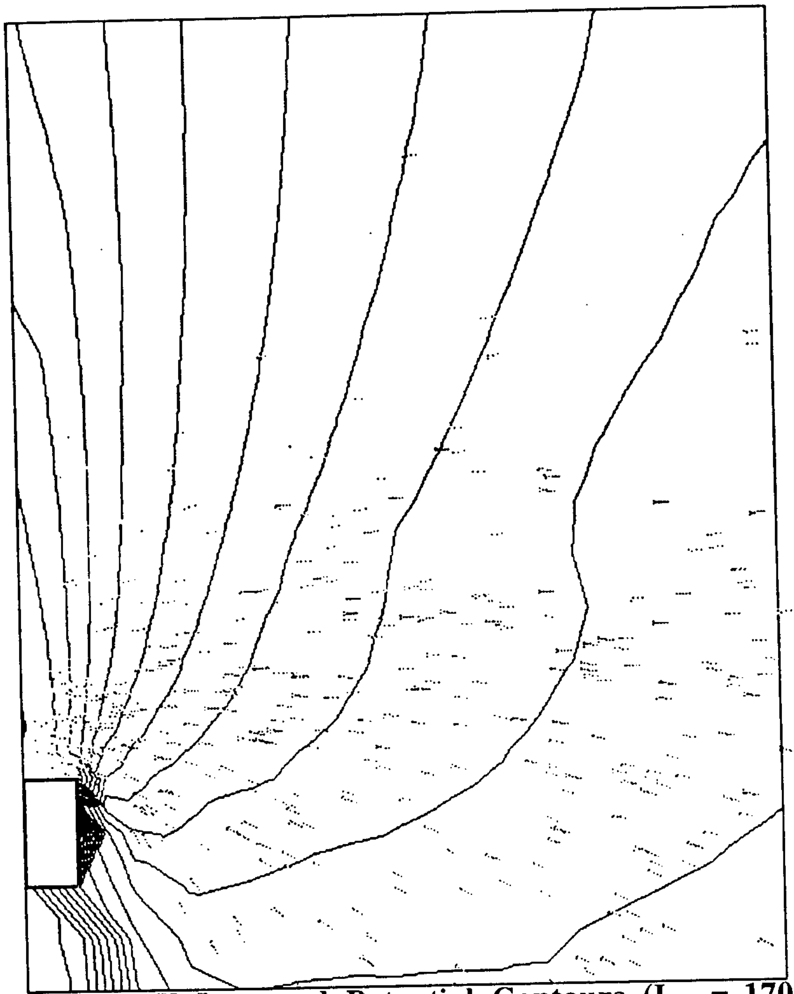


Figure 4.41: CEX Ions and Potential Contours (I_{sp} = 17030 sec.)

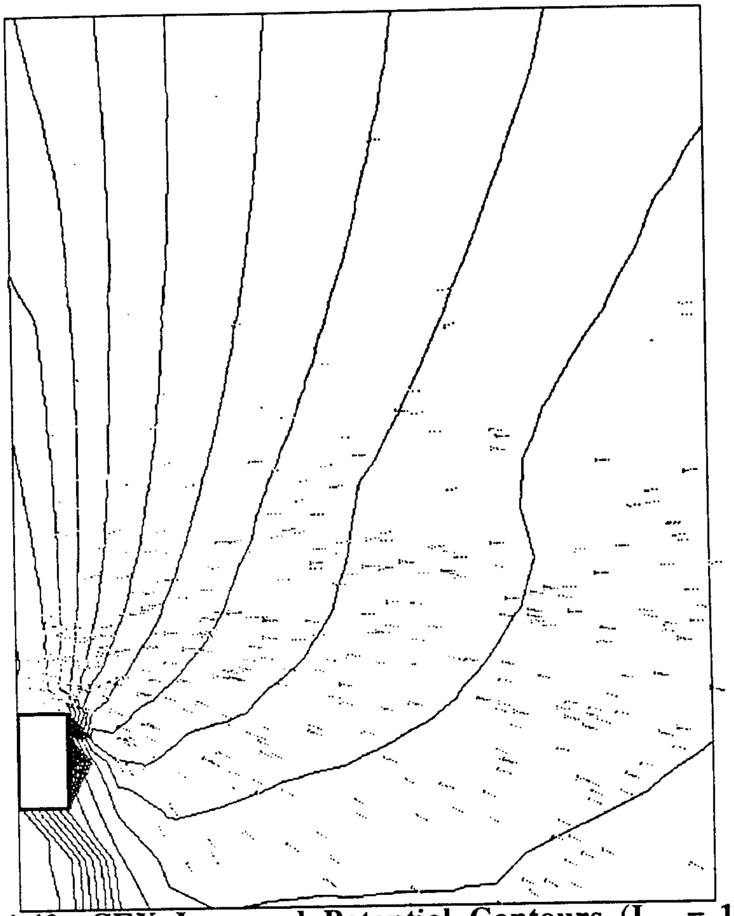


Figure 4.42: CEX Ions and Potential Contours (I_{sp} = 175 sec.)

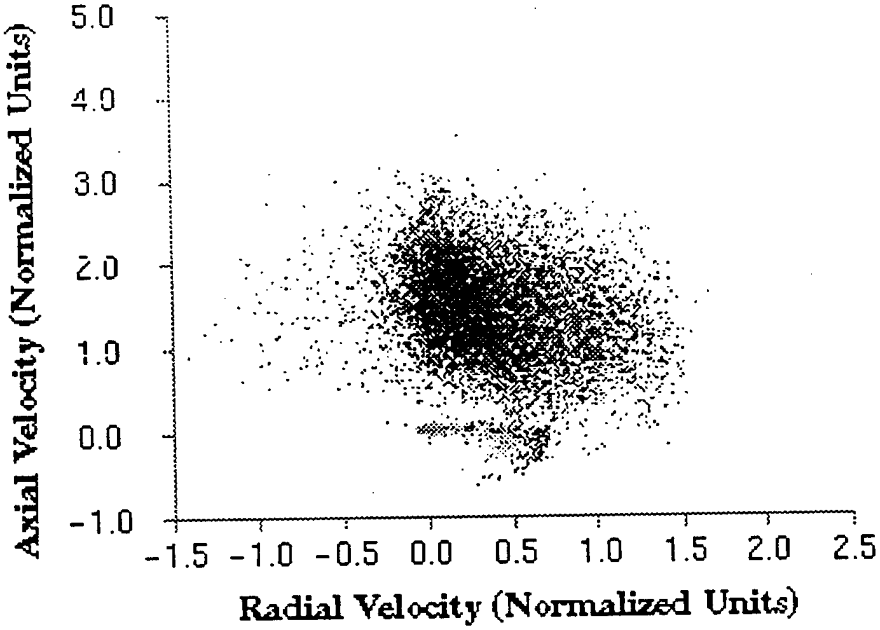


Figure 4.43: Phase Space Plot (I_{sp} = 1730 sec.)

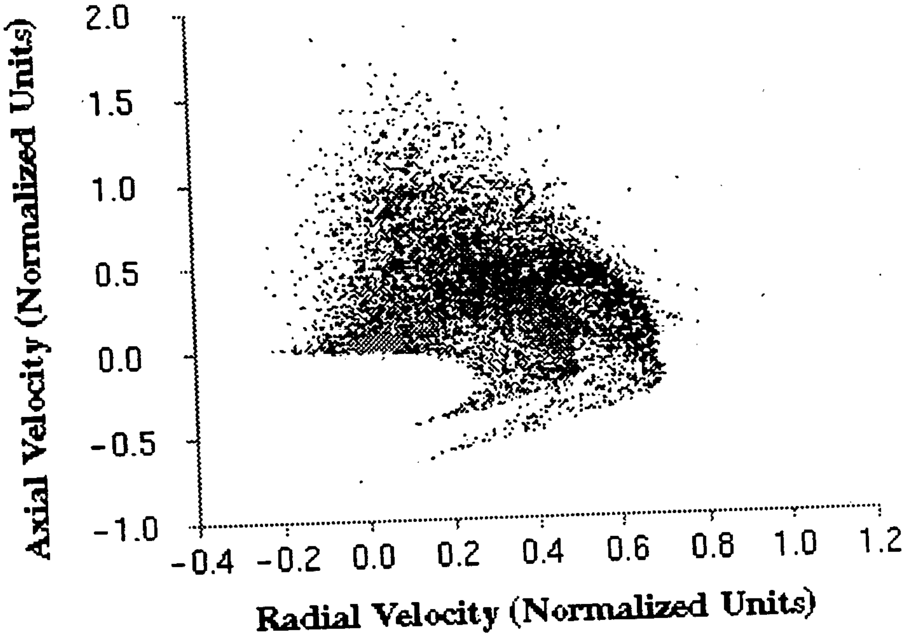


Figure 4.44: Phase Space Plot (I_{sp} = 175 sec.)

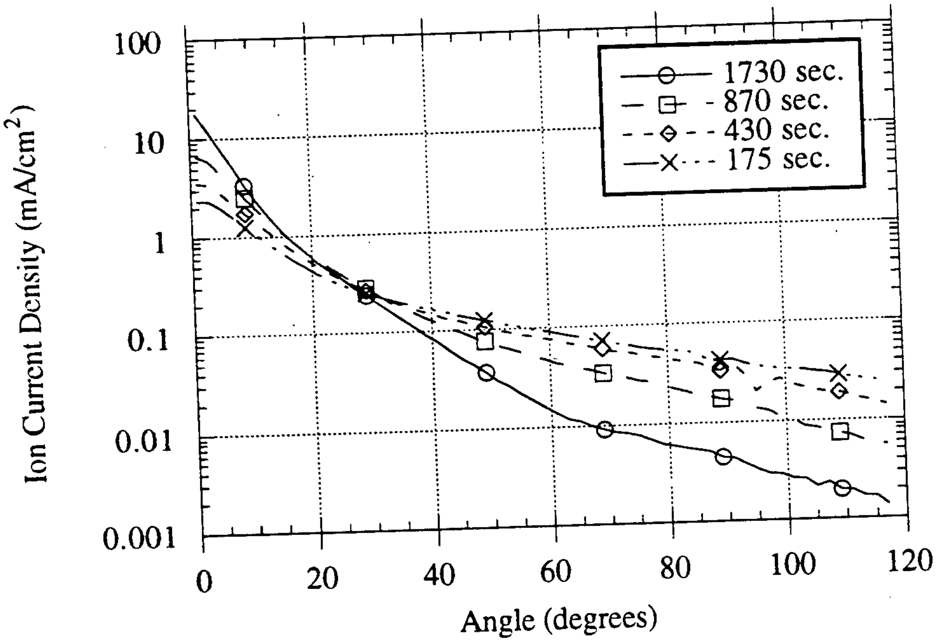


Figure 4.45: Variation in Ion Current Density with Specific Impulse

Chapter 5: Computational Plume Model and Methods (Three-Dimensional)

In Chapters 3 and 4, an axisymmetric model of a Hall Thruster plume was described in detail. A series of results were presented to verify the model against experimental data. These results included comparisons to laboratory data and predictions for a Hall thruster operating in true vacuum. These results validated the basic quasi-neutral PIC-DSMC algorithm and its surface interaction models and provided some insight into the physics of the plume region. Since real spacecraft are seldom axisymmetric, a fully three dimensional plume model is required to model the interaction between realistic spacecraft geometries and the plumes from thrusters use for orbit raising or stationkeeping applications. This chapter describes a three dimensional plume simulation named quasi3.* This simulation is based on the same basic models described in Chapter 3, but applied on a three dimensional embedded grid which can be tailored to match realistic spacecraft geometries. Because the method is very similar to the axisymmetric plume model, only those aspects which are different from the axisymmetric simulation are described in this chapter. The basic principles behind the algorithm are described in Chapter 3 and computational results are described and discussed in Chapter 6.

* For "*Quasi*-neutral *three* dimensional" plume simulation.

5.1 Introduction

The three dimensional plume simulation is very similar to the axisymmetric plume simulation and is subject to many of the same assumptions and limitations. The same quasi-neutral PIC-DSMC formulation shown in Figure 3.1 is used in the 3-D plume model. The plasma is quasi-neutral by assumption, so the simulation is valid in most regions directly impacted by the plume but invalid in wake regions behind arrays and other large surfaces. The potential is still determined by inverting the Boltzmann relationship which makes the simulation very fast in comparison to other three dimensional particle simulations. Meter scale simulations typically run in 15-24 hours on workstation class machines. This run time is thought to be short enough for effective design work. Physical quantities were normalized in the same manner as in the axisymmetric model using the quantities outlined in section 3.2.1.

A single three dimensional embedded mesh is used for both the PIC and DSMC portions of the simulation. The data structures used to store the mesh are considerably more complicated in the three dimensional model and are discussed in section 5.2. The computational domain is subject to the same external boundary conditions used in the axisymmetric model (as described in section 3.3.2). These boundaries generally reflect particles at planes of symmetry and absorb particles at boundaries to the ambient plasma. To lower computation times, three dimensional simulations are conducted without an ambient background, so no incoming particle fluxes are imposed at exterior boundaries. This is consistent with conditions in GEO, where the neutral background density is near zero [*Wang* pg. 17].

The basic PIC model used in quasi3 is very similar to that used in the axisymmetric model, and the DSMC model is virtually identical to that described in section 3.5. The same collisional phenomena are included in the three dimensional model, and since background gases are never considered in these results, Xenon-Xenon neutral collisions are always included as part of the simulation. The basic memory storage methods are also the same. Linked lists are used to store particles rather than conventional packed arrays with garbage collection. The tradeoffs between these two methods are discussed in detail in section 3.7. The major differences between the two plume models are the different density weighting factors, an integrated surface erosion and interaction model, and the inclusion of the cathode in the source model. The rest of this chapter discusses these and other differences between the two computational models.

5.2 Computational Grid and Geometry

The major motivation for developing a three dimensional plume model is to model realistic spacecraft geometries. The three dimensional PIC-DSMC model is built on an embedded Cartesian mesh. Figure 5.1 shows a typical three dimensional model of a commercial communications satellite. A simulated bus, yoke, and solar array are shown on a 3.2 m x 4.4 m x 3.2 m computational domain, and an embedded grid is visible along the edge of the main bus. This grid is collocated with the thruster and is used to better resolve the core of the plume. There are several limitations on the types of geometries which can be modeled with a 3D embedded grid. In order to improve computational efficiency, all surfaces are required to be grid conforming. This means that the satellite must be constructed from boxes that follow cell boundaries and precludes the use of curved surfaces. The primary disadvantage of this requirements it that it prevents the modeling of plume impingement on parabolic antennae. As with the axisymmetric simulation, the resolution of embedded meshes is fundamentally limited by the underlying particle simulation. Adding an embedded mesh without increasing the number of particles results in fewer particles per cell and actually reduces the accuracy of the simulation. This point is discussed in detail in section 3.3.1. The number of particles which can be added is restricted by available memory and prevents the arbitrary use of embedded grids to resolve objects of interest.

5.2.1 Computational Grid and Data Storage

The three dimensional embedded mesh is stored in a manner very similar to that used in the axisymmetric simulation. Meshes are stored in a tree structure and grids are evaluated using recursive functions. The basic grid information is stored in two data structures, one for nodes and one for cells. These structures are similar to the ones used for the axisymmetric grid and are defined using the following C code.

```
typedef struct {
  char celltype;               /* Switch to tell if cell boarders on fine cells */
  edgedata edge[6];                               /* One for each of the four edges */
  float local_time;                          /* Local time count (for MCC module) */
  float sigma_v_max;                       /* Max sigma * velocity (for MCC module) */
  int ncollision;                            /* Number of collisions in that cell */
  int summed_ncollision; /* Sum to calc. average number of collision in cell */
  unsigned int *npart;                          /* Number of particles in the cell */
  particledata **part;                      /* Pointer to list of particles in cell */
  celltosurfacemap *smap;  /* Maps cell sides to surface cells */
} celldata;

typedef struct {
  float weight;                /* Weight factor for density multiplication */
  float ex;                                   /* Electric field in x direction */
  float ey;                                   /* Electric field in y direction */
  float ez;
  float ndensity;                                   /* Neutral number density */
  float phi;                                                      /* Potential */
  float rhominus;                                  /* Negative charge density */
  float nvsamples;                      /* Running "sum" for velocity samples */
  float summed_ndensity;            /* Sum to calc. time ave. neutral density */
  float summed_rho; /* Sum used to calculate time ave. (ion) charge density */
  float summed_phi;                  /* Sum used to calculate time ave. potential */
  char switchns;       /* Switch to indicate which way to difference e field */
  char switchew;
  char switchud;                                             /* ud == Up-Down */
  short int nobj;                  /* Index to object node is assocaited with */
} nodedata;
```

Again, two structures are defined, one for cells and one for nodes. Most of the parameters present in the structures are discussed in section 3.3.1. The next two sections describe differences between the axisymmetric and 3D data structures.

Cell Data Structure

edgedata serves the same purpose in three dimensions as it does in two, but instead of four data structures, six structures are present. The term "edgedata" is actually a misnomer; each structure represents a different face of the cell (the term "edgedata" reflects the simulation's 2D heritage). The faces are referred to using the names NORTH, SOUTH, EAST, WEST, UP and DOWN as shown in Figure 5.2. Each structure contains the same components as in the axisymmetric grid structure.

celltosurfacemap is used only with cells that border on solid objects (i.e. at interior boundaries). It is a structure that maps the faces of grid cells to nodes on objects. This structure is necessary to integrate the surface interaction model with the particle simulation. Objects and surfaces are stored separately from the main embedded grid. When a particle crosses a grid cell face, the simulation checks to see if the face is an interior boundary. If it is, the simulation uses the *celltosurfacemap* to record the charge as a flux to the appropriate surface node. At the end of each timestep, the flux is used to calculate the surface potential and sputtering rates. This process is discussed in detail in section 5.6.

Node Data Structure

The 3D node data structure is almost identical to the axisymmetric node data structure. The only difference is the presence of e_z to represent the third component of the electric field and switchud to indicate the type of finite-difference formulation used to calculate the electric field in the z direction. *switchud* is a flag with the following possible values.

- CENTER indicates that a center difference formulation should be used.
- UP indicates that a forward difference formulation should be used.
- DOWN indicates that a backwards difference formulation should be used.

In all other respects, the grid formulation used is identical to that used in the axisymmetric plume simulation.

A separate grid generation program has been developed and is named *mesh3*. The following steps are used to create new three dimensional meshes.

- The top level mesh is created to define the extent of the computational domain. It contains no objects, and all grid cells and nodes are set to their default values. External boundaries are defined according to user specifications and cell and node data structures are modified accordingly.

- The child meshes are then created and cells and nodes are set to their default values. Again, external boundaries are defined by user specifications and cell and node structure modified accordingly. Cell faces that border on fine meshes are designated by setting the edge type switch equal to BORDERS_ON_FINE.
- Objects are created on the grid according to user specifications. Nodes and faces inside objects are blocked out of the simulation by designating their type as NO_BND and setting efield switches equal to NO_DIFF. Cells faces corresponding to object boundaries are designated as type INTERIOR_BND. Cell faces on interior boundaries are mapped to object surfaces using the *celltosurfacemap* data structure.

Overall, the grid generation process is relatively simple and executes in a few minutes. An arbitrarily large number of thrusters can be included in the domain. Several restrictions are present on the placement of objects relative to the edges of objects and to each other. These restrictions are generally obvious and are discussed in the User's Manual (Appendix A of this document).

5.2.2 Cell Weighting Factors

The charge density is calculated by dividing the charge collected to each node by the volume of the cell. In three dimensions, this is generally given by

$$\rho_{i,j,k} = \frac{q_{i,j,k}}{h^3} \qquad (5.2\text{-}1)$$

Before applying equation (5.2-1), it is necessary to multiply the charge collected by the ratio of the cell's total volume by the node's effective collection volume. This ratio is known as cell weighting factor. The possible weighting factors can be determined geometrically and are shown in Figure 5.3. As with the axisymmetric grid model, weighting factors are not required at embedded mesh boundaries. The weighting process is identical to that used with axisymmetric grids and is discussed in section 3.3.2.

5.3 Particle Motion

Quasi3 moves particles in the same manner as the axisymmetric plume simulation. The leapfrog method is used to integrate the basic equations of motion. In a three dimensional Cartesian system, the equations of motion (without collisions) are

$$\frac{dV_x}{dt} = \frac{qE_x}{m} \qquad \frac{dV_y}{dt} = \frac{qE_y}{m} \qquad \frac{dV_z}{dt} = \frac{qE_z}{m} \qquad (5.3\text{-}1)$$

where q/m is the charge ratio of the macroparticle. The particle velocities are calculated in all three directions and the electric field is determined self-consistently in all three dimensions using techniques described in the next section. Particles are tracked as they cross cell boundaries and the particle's location within the grid is known at all times. The three dimensional leapfrog method is subject to the same basic limitations described in section 3.4.1. In particular, the iteration timestep sets an effective upper limit on the frequency which can be accurately simulated using the quasi-neutral PIC-DSMC plume model.

5.4 Electric Field

The electric field is calculated using a procedure very similar to that used in the axisymmetric plume simulation. The basic procedure is the same. The charge density is determined as a function of position by weighting charged particles to a three dimensional Cartesian mesh. Once the charge density is known, the Boltzmann relationship (equation 3.2-4) is inverted to obtain the electrical potential. The potential is then differentiated to determine the local electric field, which is weighted back to the particle and used to solve the particle equations of motion (system 5.3-1). The main difference between the two methods is the weighting factor used to weight quantities from the grid to the particles and vice versa. The weighting functions used are first order volumetric weighting functions similar to those described in section 3.3.2, but for three dimensional Cartesian coordinates. Each particle is weighted to the eight nearest grid points at any given grid level. Given a particle at position (x_p, y_p, z_p) located between x-nodes i and i+1, y-nodes j and j+1, and z nodes z and z+1, the weighting factors used are given by the following expressions.

$$
\begin{aligned}
dx_1 &= \frac{x_{i+1} - x_p}{h} \\
dy_1 &= \frac{y_{i+1} - y_p}{h} \\
dz_1 &= \frac{z_{i+1} - z_p}{h} \\
dx_2 &= 1.0 - dx_1 \\
dy_2 &= 1.0 - dy_1 \\
dz_2 &= 1.0 - dz_1 \\
S_{i,j,k} &= dx_1 dy_1 dz_1 \\
S_{i+1,j,k} &= dx_2 dy_1 dz_1 \\
S_{i,j+1,k} &= dx_1 dy_2 dz_1 \\
S_{i,j,k+1} &= dx_1 dy_1 dz_2 \\
S_{i+1,j+1,k} &= dx_2 dy_2 dz_1 \\
S_{i+1,j,k+1} &= dx_2 dy_1 dz_2 \\
S_{i,j+1,k+1} &= dx_1 dy_2 dz_2 \\
S_{i+1,j+1,k+1} &= dx_2 dy_2 dz_2
\end{aligned}
\qquad (5.4\text{-}1)
$$

The sum of the eight weighting factors is always unity regardless of the particle's position within the cell. Once each particle has been weighted to the grid cell, the charge density is determined by dividing the total accumulated charge by the volume of the cell. The potential is then determined from the Boltzmann equation, and the electric field is determined from the potential using the same finite difference approximations used in the axisymmetric plume model (as described in section 3.6). As in the axisymmetric model, one sided difference approximations are necessary along the edges of embedded meshes. This could introduce grid artifacts to the electric field but, in practice, no discontinuities have been observed in any of the simulations.

5.5 Source Model

The source model used in the 3D simulation is very similar to the one used in the axisymmetric simulation. The same basic empirical SPT-100 model is used for flow from the anode. Ions are treated using the fits to experimental data derived in section 3.3.3 with separate thermal velocity components added in all three directions. Based on the results presented in Section 4.2.1, the higher axial ion temperature of 34 eV is used in all three dimensional plume simulations. The velocity of the particles is initially determined in cylindrical coordinates centered on the thruster and is then converted to Cartesian coordinates for use in the simulation. Arbitrary orientations can be specified for the thruster and more than one thruster can be included in the simulation. The orientation of the thruster is specified by specifying the thrust vector and a vector perpendicular to the thrust vector pointing from the center thruster towards the anode (as shown in Figure 5.4). The position of the thruster is defined by specifying the object on which it is mounted and the position of the center of the anode exit relative to that object.

The largest difference between the 3-D and axisymmetric plume models is the manner in which the cathode flow is treated. In a three dimensional geometry, this flow emerges from a cathode orifice which is placed 7.5 cm above and 1.0 cm downstream of the center of the anode exit. The simulated cathode is canted 45 degrees relative to the centerline, so the flow is directed towards the center of the plume. As with the axisymmetric simulation, cathode and anode neutrals are assumed to have a temperature of 1000 K and are choked at the cathode and anode exits respectively. Most of the ions created in the cathode are neutralized by the walls of the ionization chamber, so we assume that the ionization fraction for particles leaving the cathode is 0% (i.e. the flow consists entirely of neutral particles). The particles leaving the cathode are assumed to have a Maxwellian distribution. The particle velocity in the flow direction is given by equation (3.3-4) with the drift velocity set equal to the local sonic velocity. The velocity in the two perpendicular directions is given by a Gaussian distribution created using expressions (3.3-8) and (3.3-9).

5.6 Surface Interaction Model

One issue of particular interest to satellite designers is the interaction of the plume with surfaces of a spacecraft. In order to study surface interaction issues, a surface sputtering model has been developed and incorporated into the PIC-DSMC model. A relatively simple version of this model is used in the axisymmetric plume simulation. This model was described in detail in section 3.6. The axisymmetric model was further developed and fully integrated into quasi3 so erosion rates can be calculated on realistic spacecraft geometries. This section describes the three dimensional surface interaction model and the manner in which it is integrated into the overall plume simulation.

As discussed in section 3.6, deposition effects are ignored in the surface interaction model. This is consistent with experimental results that report little or no deposition on witness plates, even when they are placed at high angles with respect to the centerline [*Randolph* 1994]. In the axisymmetric plume model, the boundary conditions at solid surfaces are treated separately from the erosion model. In the 3D plume model, these models are combined into a single surface interaction model. When particles strike a surface, the ions and double ions are neutralized and removed from the simulation while neutrals are reflected back into the domain, as discussed in section 3.6.1. In addition, however, the incoming particle flux is used to calculate the amount of sputter damage sustained by the surface. This is done using a modified version of the sputtering model discussed in section 3.6.2.

It should be noted that surface erosion rates are post-processed in 2D so that the surface model will not slow the overall simulation. However, post processing requires that the simulation record the distribution of particles striking each surface node. While the memory required is small in 2D, it grows rapidly in 3D and was found to be too large to justify the savings in computation time. Sputtering rates are therefore calculated as the simulation runs using the following procedure.

When a particle crosses a grid cell boundary, the boundary type is determined from the cell's edge data structure (as described in sections 3.3.1 and 5.2.1). If the boundary is an INTERIOR_BND, the particle velocity, position, and grid coordinates are passed to the surface interaction model for evaluation. Each surface in the simulation is represented computationally as the side of a box. The satellite shown in Figure 5.1, for example, consists of three separate boxes, one for the bus, one for the array, and a long thin box for the yoke. Each box has six surfaces, some of which are inactive because they are located on exterior boundaries. Each active surface is divided into two dimensional cells which match the faces of the three dimensional cells in the plume simulation. The interaction model uses the object *index* and *celltosurfacemap* stored in the cell data structure to translate the particle's position in the computational grid into a cell on the face of one of the objects in the simulation. The particle's charge and/or particle weighting factor are recorded as striking that particular surface cell. The system is similar to the nearest neighbor method of particle weighting used in zeroth order PIC simulations. Because the surface grid is cell centered, the particle is weighted straight to the cell and no geometric comparisons or weighting factors are required.

Calculating the sputtering damage caused by the impact of a particle requires a knowledge of the particle's impact energy. The impact energy depends in turn on the potential drop across the non quasi-neutral sheath at the surface of the object. When creating an object, it is necessary to specify its potential with respect to the plasma. Two difference types of objects can be specified: objects of fixed potential and objects which float with respect to the plasma. A solar panel, for instance, could be represented as an object of fixed potential. Each node on the array could be assigned a different potential to mimic the distribution of cells across the array. A kevlar thermal blanket, on the other hand, would be represented by an object of floating potential. These objects are treated as dielectrics whose potential can be determined by balancing the flux of ions and electrons to each node independent of the nodes which surround it. A sheath interaction model is used to determine the surface potential at the end of each timestep. This potential is recorded for use in the sputter erosion model. The sheath interaction model for floating surfaces is similar to the sheath model described in section 3.6.1. Since objects float at potentials which are negative with respect to the ambient plasma, the Bohm sheath criterion requires that ions entering the sheath must be moving faster than the ion acoustic velocity. Once an ion enters the sheath, it is continuously accelerated until it strikes the wall, and the sheath/pre-sheath boundary therefore represents a sink into which ions disappear and never reemerge. The potential can therefore be determined using equation (3.6-1), i.e.

$$\Phi_f = -\frac{kT_e}{e}\ln\left(\frac{4\Gamma_i}{n_o\bar{c}_e}\right) \qquad (3.11\text{-}1)$$

The flux used in (3.6-1) is the cumulative flux seen by the surface over the course of the entire simulation. This lowers the noise in the final result. Sampling typically starts after the first 1000 timesteps to lessen the influence of the startup transient.

For surfaces of fixed potential, the sheath conditions are more complex. When the surface potential is less than the local plasma potential, the pre-sheath can not "see" particles in the sheath region and the magnitude of the potential drop is irrelevant. When the surface potential is greater than the plasma potential, the electric field is reversed and the sheath reflects ions away from the wall. Since the flux of electrons to a surface is generally much larger than the flux of ions to the surface, this is unlikely to occur. Small sections of the solar array may end up sitting positive with to respect to the plasma, but the bulk of the spacecraft will sit negative instead. Because a positively floating surface attracts electrons, the electrons near the wall will not be in equilibrium. As a result, it is impossible to correctly model the electron dynamics and the potential structure in the sheath and pre-sheath region within the using a Boltzmann electron model. Nevertheless, we choose to model the electrons using the Boltzmann relationship.

Fortunately, it is possible to model the dynamics of ions near positively charged surfaces. When ions strike a positively charged surface, they are assumed to cross the sheath/pre-sheath boundary. Some of these ions are reflected by the sheath and others are neutralized by impact with the surface. The potential rise across the sheath is calculated by subtracting the local plasma potential from the fixed surface potential. The energy of the incoming particle is then compared to the sheath potential. If the particle's total kinetic energy is greater than the sheath potential, the ion's impact is recorded and it is removed from the simulation. If the particle's energy is less than the sheath potential, the particle's impact is not recorded and the ion is reflected back into the domain. The net result is that low energy ions are reflected while high energy ions are neutralized by the underlying surface. This accurately reflects the behavior of ions when they encounter a positively charged surface. Only the particles which strike the surface can cause sputtering damage. Neutrals are unaffected by the presence of the sheath and always cause sputtering damage. As with the floating boundary, this surface model is only valid when conventional 1D sheath theory applies. This is generally the case when the Debye length is much less than the length scale of the surface, which is a restatement of the quasi-neutral assumption. The implications and limits of validity for this assumption are discussed in detail in section 3.6.

Sputtering rates are calculated by tabulating the material removed by each ion and neutral which strikes the surface. The amount of material lost is determined by multiplying an energy dependent sputtering coefficient by the macroparticle weighting factor, i.e.

$$V = S(E_k)W$$

Where E_k is the total kinetic energy of the particle striking the surface. The erosion depth is determined by dividing the volume by the area of the cell and the erosion rate is determined by dividing the rate by the total simulation time. The total erosion volume is the sum of the amount eroded by each particle over the course of the simulation and multiple materials are simulated by tracking several independent erosion volumes. A particle's impact energy is given by the sum of its initial kinetic energy and the energy it gained or lost in the sheath. All particles are assumed to enter normal to the surface and neutrals undergo no acceleration in the sheath region.

Three different materials were considered in this thesis when calculating sputter erosion rates. Two of the materials were also considered in the axisymmetric plume model. The first material is Silver, which is commonly used for solar cell interconnectors. Its energy dependent sputtering coefficient is given by the linear fit shown in equation (3.6-2). The second material is Quartz, which is used to model solar cell coverglasses. Its sputtering coefficient is also given by equation (3.6-3). The third material modeled was Silicon. Its sputtering coefficient was derived from laboratory data collected by *Rosenberg* [1962] and shown in Figure 5.5. Also shown is a linear fit based on the following expression

$$S = 1.0577 \times 10^{-3} E - 0.12115 \qquad (5.6\text{-}1)$$

In all cases, the sputtering coefficients used are for Xenon ions striking normal to the surface. As discussed in Chapter 3, atoms striking at different impact angles have been observed to have higher sputtering coefficients [*Chapman* pg. 247]. At the present time, however, the angular dependence for Xenon sputtering Silver, Quartz, and Silicon is unknown and can not be included in the surface interaction model. As a result, this surface model may underpredict the actual erosion rate. Better experimental measurements of the sputtering coefficient are needed to improve the sputtering coefficients used in the surface model and include the effects of different ion impact angles.

5.7 Summary

In summary, a three dimensional PIC-DSMC simulation has been constructed based on the observation that the plume is a quasi-neutral, unmagnetized plasma in which the electrons are effectively collisionless. The basic algorithm is the same one used for axisymmetric simulations, and a variety of different collisional phenomena are included in the model. An SPT-100 thruster is simulated using an empirical source model which is based on experimental data. This primary difference between this model and the axisymmetric model is that the presence of the cathode can be directly modeled in a 3D geometry. A surface interaction model has also been developed for the 3D simulation. This model is fully integrated into the plume simulation and predicts the erosion rate for materials placed in the plume region. The final simulation was written in ANSI C, and simulations were conducted on Digital, IBM, and Hewlett-Packard workstations with 96-256 Megabytes of RAM. Run times were typically 12-15 hours for the three dimensional geometries. A major advantage of the three dimensional model is that it runs significantly faster than previous models and can run on a workstation class computer. The relatively fast turn around time may allow spacecraft designers effectively evaluate realistic spacecraft designs.

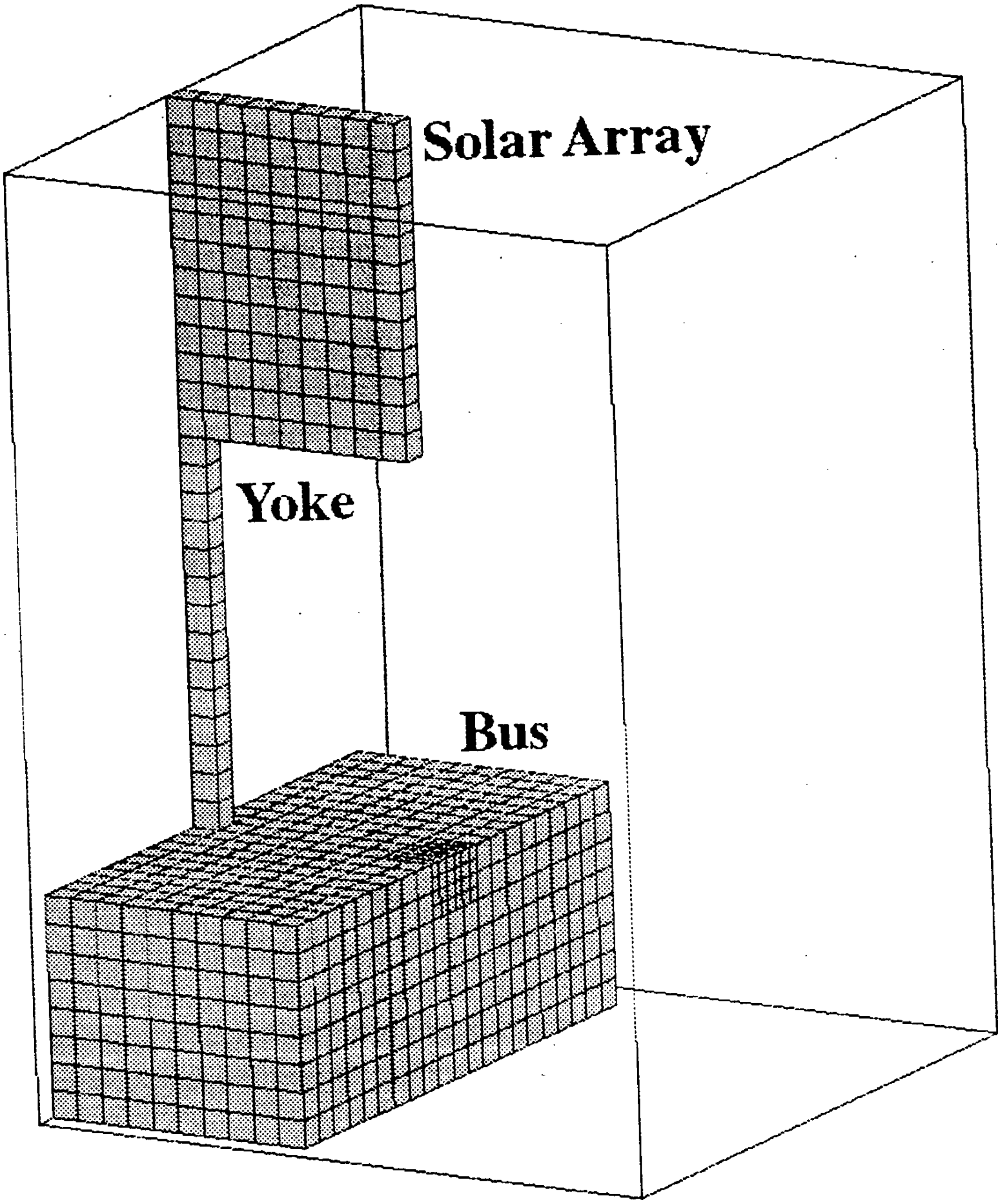


Figure 5.1: Model of a Communications Satellite on a 3D Embedded Mesh

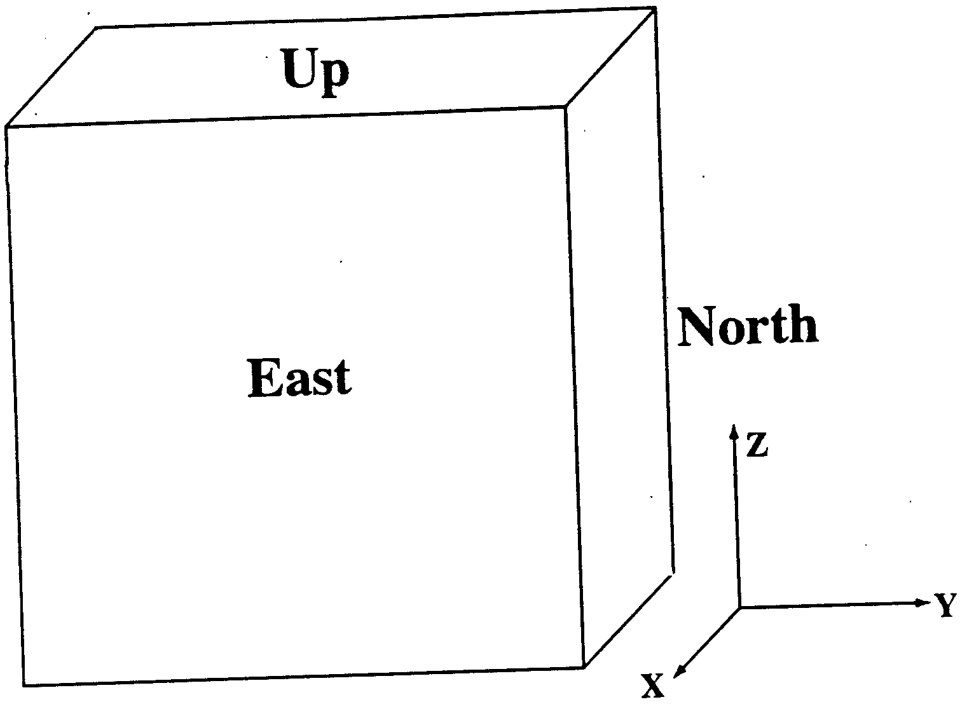


Figure 5.2: Nomenclature on Three Dimensional Grid
The West, South, and Down faces lie directly across from the East, North, and Up Faces

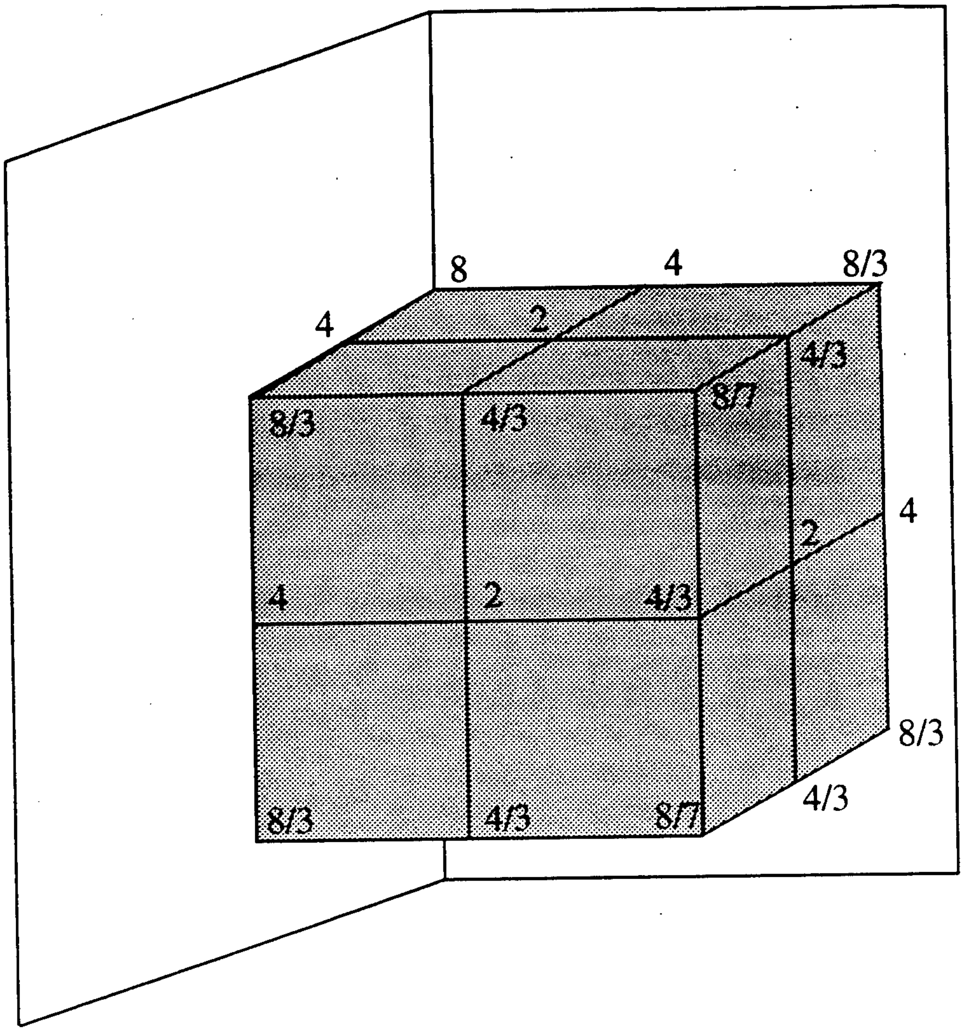


Figure 5.3: Weighting Factors in Three Dimensions
Fractions indicate weighting factor.
Weighting factor = 1/(fraction of volume around node from which charge can be collected)

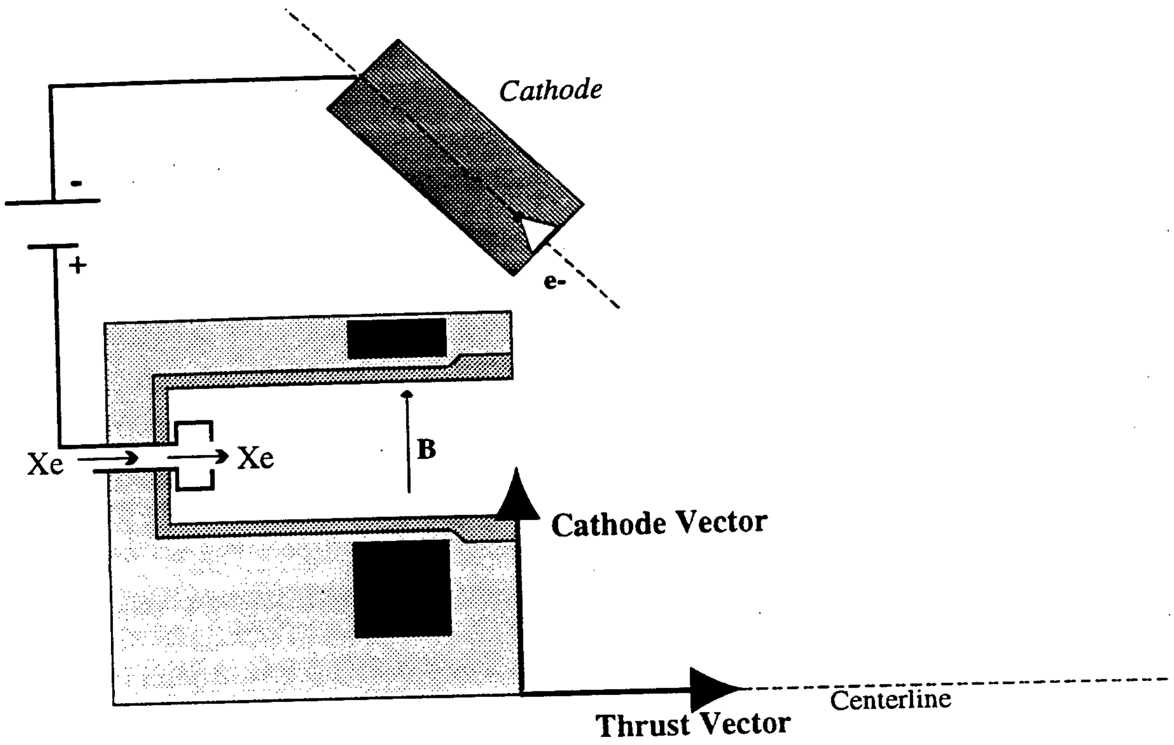


Figure 5.4: Vectors Specifying Orientation of Hall Thruster

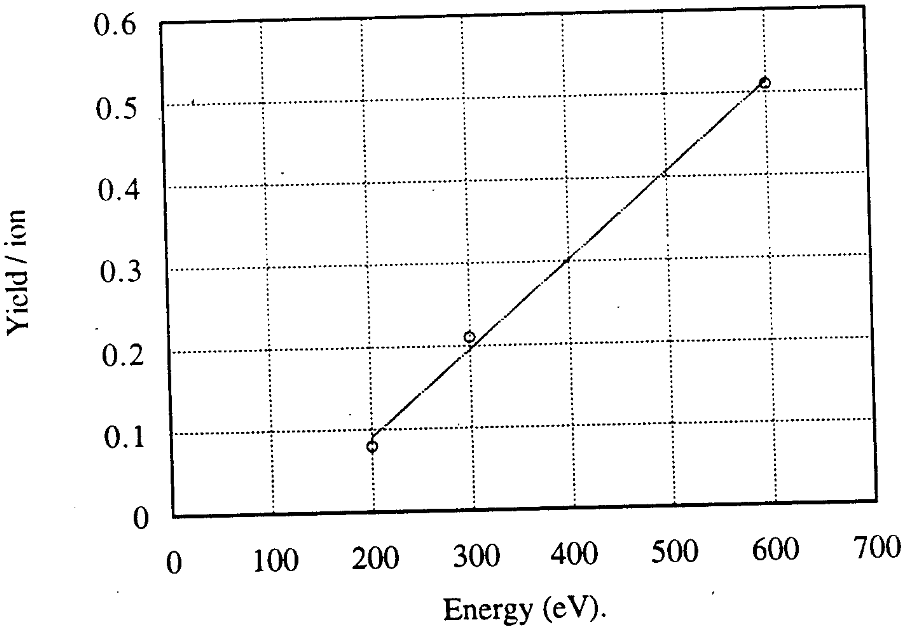


Figure 5.5: Sputtering Coefficient for Silicon, Measured and Fit
[Rosenberg, 1962]

Chapter 6: Three Dimensional Results and Discussion

A three dimensional PIC-DSMC model of an expanding plasma plume has been constructed and used to simulate the plume of a Hall Thruster. The simulation is named *quasi3* and is used to produce three dimensional simulations of Hall Thrusters operating in space, on the ground, and as part of a realistic model of a geosynchronous communications satellite. The model has been verified against experimental data and against results from the axisymmetric plume simulation. This chapter presents results from the three dimensional plume simulation *quasi3* and discusses them in some detail. Section 1 shows results of simple comparisons to experimental data and to results from the axisymmetric simulation and section 2 presents the results of simulations of a thruster mounted on a typical geosynchronous communications satellite.

6.1 Comparisons to Experiment and to Axisymmetric Results

As was discussed in Chapter 5, *quasi3* is based on the same quasi-neutral PIC-DSMC model used in the axisymmetric plume model. This algorithm was validated in the axisymmetric geometry through direct comparison to experimental data. Once the basic method was verified and understood, the model was expanded to three dimensions for use in simulating realistic spacecraft geometries. To check if any bugs had been introduced during the conversion from the axisymmetric geometry to three dimensions, *quasi3* was verified through comparison to the same ion current density data shown in Chapter 4. Comparisons were also made to the results from the axisymmetric simulation. This not only helps to isolate problems in the code, but also verifies that our results are geometry independent.

The geometry used to verify quasi3 is relatively simple and is shown in Figure 6.1. A single thruster is mounted on a box in a three dimensional Cartesian domain. The dimensions of the domain are 2.5 m x 2.5 m x 1.6 m in the x, y, and z directions respectively. The box on which the thruster is mounted is a cube 0.2 m long in each dimension. A single embedded grid is placed in front of the thruster exit. It has dimensions of 0.5 m x 0.5 m x 0.4 m. Each grid cell in the domain is 10 normalized units wide, so the top level mesh is 24 cells x 24 cells x 15 cells in dimension. To facilitate comparisons to experimental and axisymmetric data, the ion current density in the simulation was measured using a "virtual surface" similar to the one described in Chapter 3. The flux of ions passing though a hemisphere centered on the thruster exit (typically with a radius of 0.6 m or 1.0 m) was recorded and used to determine the ion current density as a function of angle from the centerline. The flux recorded was an average flux measured in all directions, so asymmetries in the plume do not appear in the ion current density results. This was necessary to ensure that enough particles were sampled at all angles to give statistically valid results. The presence of background gas was simulated using the same techniques used in the axisymmetric simulation (as described in Section 4.2). Each simulation was run for 3000 timesteps using a timestep of 0.1 normalized units. This allowed the simulation to reach steady state in which the ion current density distribution no longer varied with the number of iterations. Unless otherwise stated, each individual run took from 12-20 hours of run time on a Digital Alphastation, IBM RS 6000 workstation, or HP workstation with 96-256 megabytes of memory..

Figures 6.2, 6.3, and 6.4 show several plots of the ion current density 60 cm from the thruster exit at pressures of 2.2 x 10^{-6} Torr, 2.5 x 10^{-5} Torr, and 6.3 x 10^{-5} Torr respectively. Three types of results are overlaid on each graph: a set of axisymmetric results (marked as "2D"), a set of three dimensional results, and experimental data from *Manzella* [1995]. The simulated results agree very well with each other and confirm that the change from an axisymmetric to a three dimensional geometry has not significantly affected the simulation. Both sets of results show good agreement with data, again matching to within a factor of 2-3 across the domain. The very high level of agreement between the axisymmetric and 3D results across a range of pressures verifies that the two simulations are modeling the same basic physics and giving the same basic results.

Simulations were also made of the SPT thruster operating in vacuum. Figure 6.5 is a contour plot of the charge density in the plume region. The plume itself is quite symmetric, and does not exhibit three dimensional effects. However, Figure 6.6 shows the neutral density in the plume region. The neutral density distribution is clearly skewed to one side of the plume. This asymmetry is caused by the presence of the hollow cathode at the right of Figure 6.6. A substantial fraction of the neutrals in the plume are emitted by the hollow cathode and are directed towards the centerline. They form an asymmetric distribution in space which then interacts with the ions in the plume through CEX collisions. Although the neutrals also have an non-symmetric velocity distribution, the ions in the plume still form a symmetric structure. The reason for this is apparent in Figure 6.7. Figure 6.7 is a phase plot of the x and z velocities of particles in the plume region. The neutral velocity is generally much less than the velocity of ions in the plume. As a result, CEX ions pick up most of their velocity from the electric field in the plume region. As was discussed in Chapter 4, most CEX collisions occur in a relatively small region directly in front of the thruster and near the thruster's centerline. Since the electric field is generally axisymmetric with respect to the thrust vector, the CEX ions are accelerated in all directions and rapidly acquire a nearly symmetric velocity distribution. This produces a charge distribution which is symmetric in space even through the neutrals have an asymmetric distribution.

An interesting numerical observation is that the number of particles per cell varies very differently in 3D and axisymmetric particle simulations. In an axisymmetric simulation, if a gas has a uniform density, the number of particles per cell is proportional to the square of the radius. If the density in the plume falls as $1/r^2$, the two effects cancel and the number of particles per cell is approximately equal across the domain. In a 3D particle simulation, if a gas is of uniform density, the number of particles per cell is constant. Therefore, if the density of a plume falls as $1/r^2$, the number of particles per cell also falls as $1/r^2$. This creates problems with trying to maintain a minimum number of particles per cell throughout the domain. In the results shown in Figure 6.2-6.7, the number of particles per cell varies drastically from 1600 per cell in the core of the plume to less than 5 per cell at the edges of the domain. Traditionally, both PIC and DSMC schemes require 10 particles per cell to retain statistical accuracy. However, when the 3D results created with 3-1600 particles per cell are compared to axisymmetric results produced with 20-2000 particles per cell, the results are virtually identical. Note that the cells with the fewest particles in the axisymmetric simulation are near the centerline while those with the fewest particles in the 3D simulation are near the edges of the plume region. The fact that the axisymmetric and 3D results correlate so well indicates that having less than 10 particles per cell at the edges of the plume does not affect the accuracy of the simulation. This is believed to be the case because most of the important physics is occurring in the center of the plume, where the majority of the CEX collisions are created and accelerated towards the edges of the plume. Particles at the outside edges of the plume, on the other hand, undergo relatively few collisions and follow almost ballistic trajectories. Based on these results, we conclude that the 3D particle simulation is fairly robust with respect to the number of particles per cell in the domain. Therefore, while we maintain a large number of particles per cell in the core of the plume (hundreds to thousands), we exercise some latitude in the outer parts of the 3D simulation. In cases, fewer than 5 particles per cell are maintained at the edges of the computational domain. To determine the sensitivity of the simulation to the number of particles, a simulation was run with a very large number of particles on the geo-spacecraft configuration discussed in the next section. As we shall see below, the number of particles had no effect on results produced by the simulation).

6.2 Simulation of Realistic Satellite Geometry

The main benefit from developing a three dimensional simulation is that it can be used to evaluate more realistic geometries than can be considered in an axisymmetric simulation. At the present time, one of the applications that designers are most interested in for the SPT-100 thruster is in North-South stationkeeping for geostationary communications satellites. Thrusters on a satellite of this type would be oriented on the North and South faces of a satellite (as shown in Figure 6.8) and fired in groups to apply Δv without changing the satellite's orientation with respect to the Earth. *Quasi3* was used to study the effect that the plume of an SPT-100 thruster would have on a satellite when used for North-South stationkeeping.

The study was conducted on a generic geo-comsat geometry loosely based on an Intelsat satellite configuration. This configuration is shown in Figure 6.9. The satellite configuration shown consists of a Bus, Yoke and Solar array located on a 3.2 m x 4.4 m x 3.2 m domain discretized into 25 x 35 x 25 cells. Each cell is 12 normalized units (0.126 m) long on each side. The section of bus shown has dimensions of 1.1 m x 1 m x 2.6 m and represents a quarter of the spacecraft's main bus. A 1.9 meter yoke connects the bus to the end of a solar array. The array is 1.14 meters wide and continues for up to 14 m off the top of the domain. Only the bottom 1.5 meters of the array are included in the simulation because this area it thought to experience the most degradation due to plume interaction effects. The SPT-100 thruster is mounted at the edge of the bus under the edge of the solar array as shown in Figure 6.9. The thruster's orientation is varied during the simulation based on two parameters: a cant angle and an array angle. These angles are defined as shown in Figure 6.10. The cant angle is the angle between the thrust vector and the vertical axis. Lower cant angles better orient the thrusters for N-S stationkeeping and higher cant angles cause an effective loss of I_{sp}. The array angle is defined as the solar panel's angle relative to an imaginary line connecting the yoke to the thruster. On a real spacecraft, the position of the thruster is fixed and the array turns with respect to the thruster. In our model, the requirement that objects be grid conforming means that the array can only be turned in 90 degree increments. Rather than trying to simulate an array turned 45 degrees with respect to the thruster, we simulate intermediate array angles by moving the thruster with respect to the array. This is shown, for instance, in Figure 6.11 where the thruster is placed at a position 45 degrees away from the array to simulate an array turned 45 degrees with respect to the thruster. This allows us to effectively simulate a rotating array and still create a grid conforming bodies. As the array angle increases to 45 degrees, the end of the plume tends to get cut off by the side of the domain in the z direction. To avoid cutting off the end of the plume, the size of the computational domain was increased from

(25 x 35 x 25) to (25 x 35 x 35) cells. This provides the plume with enough room to expand and develop, as shown in Figure 6.11.

All simulations were run for 15000 time steps with a time step of 0.1 normalized units. Each individual run took from 12-20 hours on a Digital Alphastation, IBM RS 6000 workstation, or HP workstation with 96-256 megabytes of memory. The potential of the entire surface of the array was fixed at -92 volts relative to the center of the plume. A real array would be covered by cells with potentials that vary from 0 to 92 Volts. Setting the entire surface to -92 V therefore represents a worst case in which all parts of the array are assumed to sit at a relatively negative potential with respect to the spacecraft. The proper way to determine the potential of the array would be to determine the potential of the spacecraft as a whole relative to the plasma. This would require "closing the loop" on the current collected and emitted by the spacecraft as a whole, which introduces a new set of time and length scales to the problem. Fully closing this loop has never been done and is probably beyond the capability of current computing workstations. We therefore choose to fix the arrays at a set potential.

A cant angle of 45 degrees and an array angle of 45 degrees were chosen as the spacecraft's baseline configuration. Figure 6.11 shows this configuration overlaid by a surface of constant potential. The plume is clearly visible as a cone emerging from the top of the bus and oriented away from the solar array. Figure 6.12 shows a contour plot of ion current density on the face of the array nearest to the plume. Although the plot is noisy, a small but noticeable flux of ions is clearly reaching the surface even though the thruster is oriented away from the array. As one would expect, the area of highest flux is in the corner of the array which sits closest to the anode exit. Very little current reaches the lower or upper left corners of the array. Figure 6.13 is a contour plot showing the energy of ions striking the surface of the array. It shows that the ions with relatively high energies are actually striking the upper right corner of the array, while ions with relatively low energy strike the corner nearest to the thruster. This occurs because high energy ions follow relatively straight trajectories and do not turn far enough to strike the lower part of the array. CEX ions, on the other hand, have a small turning radius and are easily influenced by electric fields at the edge of the plume. These ions turn quite sharply and end up striking the bottom of the array.

As discussed in section 6.1, fewer than 10 particles per cell are present in many parts of the domain, and as little as 3 particles per cell are present at the edges of the domain. This is less than is traditionally permitted in PIC and DSMC simulations. To evaluate the impact that the number of particles has on the results from the simulation, a simulation was carried out in which the number of particles in the simulation was quadrupled across the domain. This simulation took several days to complete and required 256 megabytes of memory. Figures 6.14-6.16 shows the estimated ion current and energy distribution of ions in the plume and striking the face of the solar array. The results in Figure 6.14 are virtually identical and impossible to distinguish from each other. The results in Figures 6.15 and 6.16 are much smoother than those in shown in Figures 6.12 and 6.13. However, the two sets of results are otherwise very similar to each other. On the basis of this comparison, we conclude that the results are insensitive to the number of particles used in the simulation, even though fewer than 10 particles per cell are present in some parts of the domain. This is believed to be the case because most of the important physics is occurring in the center of the plume, as discussed at the end of section 6.1.

Figures 6.17, 6.18, and 6.19 show the calculated erosion rates for Silver, Silicon, and Quartz surfaces of the array in the baseline configuration. All materials are assumed to sit at -92 volts with respect to the center of the plume. These figures show that a noticeable and potentially significant amount of erosion will occur to interconnectors and coverglasses on the solar array. As one would expect, the highest erosion rates occur on surfaces closest to the thruster. Silver in particular has an erosion rate greater than 1 micron per month in some parts of the array. Quartz glass shows the next highest erosion rate, about 0.4 microns per month, and silicon is the most sputter resistant material, with a maximum rate of only 0.142 microns per month. The actual area over which these high rates occur is relatively small, covering an area of no more than 0.25 m^2. In addition, it should be noted that these surfaces are being held at a negative potential with respect to the plume. One obvious way to lower the peak erosion rate is to bias cells at the corners of the array positive with respect to the spacecraft. This would lower the energy of ions striking the surface of the array and help mitigate sputtering losses. Another approach would be place a plume shield near the plume to deflect ions away from the array. This approach might be quite effective, though it could also incur substantial mass penalties.

Canting the thruster away from the satellite's N-S axis lowers the thruster's effective I_{sp}, so it is desirable to use as small a cant angle as practical. To investigate the effects of changing the cant angle, simulations were run with thruster cant angles varying from 0 to 45 degrees. Figure 6.20 shows the simulated erosion rate of silver on the same surface shown in Figure 6.17, but with the thruster canted at only 30 degrees. The 15 degree change of angle has a significant impact on the erosion rates on the array. Not only is the peak sputtering rate more than 3 times higher than in Figure 6.17, but the area over which damage occurs extends much farther up the side of the array. A plume shield would almost certainly be required to limit damage to the array. Figure 6.21 shows the sputtering rate of silver on the same surface with a cant angle of zero degrees. The erosion rate is now over an order of magnitude higher than in Figure 6.17, and is clearly unacceptable. The area of maximum damage now extends up the array to the end of the computational domain. The damage would certainly extend further if the domain continued to the end of the array. These results show the necessity of canting EP devices away from the NS axis of the spacecraft to prevent damage to the solar arrays. Although this canting is required to prevent damage to the array, it is also lowers the thruster's effective specific impulse by the cosine of the cant angle. This loss must be considered when using these thrusters on the spacecraft.

As the satellite orbits around the Earth, the array rotates about the satellite's N-S axis so it will always point directly at the Sun. This changes the orientation with respect to the thruster by changing the array angle. Unlike chemical thrusters, which undergo relatively short, impulsive burns, Hall Thrusters must operate for many hours at a time to maintain a satellite's position in orbit. The thruster will therefore see the array from a range of different orientations during the course of its operation. To examine the effect that the array angle has on the interaction between the plume and the surface of the array, simulations were also carried out over a range of different array angles. The different cant/array angle combinations simulated are shown in Table 6.1.

Cant Angle	Array Angle
0	0
0	15
0	45
0	90
15	0
15	15
15	45
15	90
30	0
30	15
30	45
30	60
30	75
30	90
45	0
45	15
45	30
45	45
45	60
45	75
45	90

Table 6.1: Cant/Array Angles Simulated in 3D
(All entries in units of degrees)

Figure 6.22 shows the spacecraft configuration used to simulate array angles greater than 45 degrees. The array has been turned 90 degrees with respect to Figure 6.9 so the array angle shown is 90 degrees. The array is the same size as that shown in Figure 6.9, but now the entire width of the array is located on the domain. Figures 6.23 and 6.24 show the simulated ion flux and silver erosion rates on the array at a cant angle of 45 degrees and an array angle of 90 degrees. The area shown is twice as large as that shown in Figure 6.12. As one would expect, the ion flux is lower than that shown in Figure 6.12 because the array is farther away from the thruster exit. The area of maximum ion flux and sputter damage is in the bottom center of the array, which is the portion of the array closed to the thruster. The dark spot in the center of the bottom axis is due to the incorrect weighting factor used at the point where the array intersects the yoke (see Chapter 5). The observed erosion rates are smaller than those seen in Figure 6.12, but are still significant. However, the damage occurs is a different part of the array, so the damage seen is not cumulative. Overall, the lower corner of the array appears to be the point most exposed to the plume.

Figure 6.25 shows the ion flux to the array at a cant angle of 30 degrees and an array angle of zero degrees. Although the lower left corner of the array sits squarely within the plume, virtually no ions strike anywhere on the array. This occurs because the ions leaving the thruster are moving nearly parallel to the surface of the array so their trajectories seldom intersect the surface. This result is not physically accurate and reveals some of the limitations of the quasi-neutral PIC method. When the array angle is zero, the geometry is similar to that of a charged plate at zero angle of attack in a hypersonic plasma flow. This special case has been described in detail by Wang [1991]. Wang identified three flow regimes in the vicinity of the plate: a leading edge region (region I), a fully developed sheath layer (region II) and the main quasi-neutral flow (region III). These regimes are shown in Figure 6.26. If the sheath is thin compared to the scale of region III, the plate acts as a sink to the quasi-neutral ion flow. This results in the formation of an expansion fan centered at the leading edge, similar to the Prandtl-Meyer expansion of supersonic gas around a convex corner Wang [1991]. The ion current collected by the plate is therefore independent of the surface potential and has a uniform density given by

$$j(\text{region II}) \equiv 0.37 e n_o C_s \qquad (6.2\text{-}1)$$

where c_S is the ion acoustic velocity. For a thruster at a cant angle of 0 and array angle of 0, the resulting current is about 2 x 10^{-4} mA/cm^2, which at the limit of the currents that can be resolved in our simulations. The result is therefore consistent with Figure 6.24. However, expression (6.2-1) is not valid in the leading edge region where the expansion fan has not yet had time to develop. The boundary between region I and region II is at a distance L_2 from the leading edge of the plate. From Wang, L_2 is given by

$$\frac{L_2}{\lambda_d} = 1.3 M_o \left|\hat{\phi}\right|^{0.75}$$

Where M_0 is the free stream ion mach number and $\hat{\phi}$ is the normalized wall potential. The array is fixed at a potential of -96 V relative to the core of the plume, so L_2 is about 1 m when the cant angle and array angle are both zero. Since the section of array simulated is only 1.5 m long, the bottom two thirds of the simulated array acts as the leading edge. Wang showed that the flux to the leading edge can be many times higher than the flux given by (6.2-1). This effect is not captured in our simulation because the increased flux is a non-quasi-neutral effect. The simulation results are therefore invalid when the array angle is zero and it would be unwise to operating the thruster when the array angle is zero. Results for other array angles are not affected. The simulation is valid on the front of the array, where the plume directly impinges on the array, but invalid in the low density wake region.

Spacecraft designers are most interested in the relationship between the cant angle, array angle, and the erosion rates experienced on the array. A cant angle of zero degrees is clearly unacceptable, for instance, while a cant angle of 45 degrees is probably quite acceptable in most array orientations. Simulations conducted at different cant and array angles can be used to produce a map of the relationship between the array angle, cant angle, and erosion rates on the array. Since it would take a long time to completely fill out the parameter space, we have taken results from the runs listed in Table 6.2 and extrapolated to different cant and array angles using weighted averages based on 1/distance squared weighting factors. Figures 6.28 and 6.29 show the results of these calculations. Figure 6.28 shows the erosion rate experienced by the lower right corner of the solar array (i.e. the part of the array closest to the thruster when the array angle is 90 degrees). Figure 6.29 shows the erosion rate experienced at a point 20 centimeters from the bottom of the array and 20 centimeters from the side of the array, or about 30 cm away from the lower right corner. The values on this graph are extrapolated from a total of 21 data points as shown in Table 6.2. The upper left corners of these graphs are significantly undersampled, but this regime is not generally of interest to spacecraft designers. As one would expect, lower cant angles and array angles result in significantly higher erosion rates. In fact, the erosion rate varies by more than 2 orders of magnitude over the parameter space.

It is clear that silver interconnectors will undergo some erosion over the lifetime of the satellite. The allowable cant and array angles for this configuration depend on the acceptable erosion rate and the amount of time the array will be exposed to the plume from the SPT thrusters. Determining these rates requires some assumptions about the satellite's configuration and duty cycle. As a baseline case, we assume that the thruster is being used for N-S stationkeeping on a 2000 kg. satellite in GEO with a lifetime of 12 years. The total Δv required over the lifetime of the satellite is about 617 m/s [*Larson and Wertz* pg. 151]. We also assume that a total of four thrusters are mounted on the satellite, two on each side as shown in Figure 6.27. Each SPT-100 thruster produces 85 mN of thrust, but because the four thrusters are canted with respect to the array, this value must be multiplied by the cosine of the cant angle. The highest cant angle considered is 45°. In this orientation, the SPT-100's effective thrust is 60 mN. This represents the worst case thrust, and we use it for the rest of this analysis. The total thruster operation time can be calculated using the following relationship.

$$t = \frac{m\Delta v}{F}$$

The thrusters would probably be fired in pairs, so the effective force on the satellite is 120 mN. The resulting operation time is 2852 hours, or 4.0 months of continuous operation time. The solar array orientation changes continuously as the satellite travels in its orbit around the Earth. Only one side of the array is sensitive to erosion and, as shown in Figure 6.27, while half of the array sits at an array angle less than 90° while the other half of the array sits at an array angle greater than 90°. Since the simulated erosion rates are very small at array angles greater than 90°, each half of the array will, on average, be exposed to the plume only half of the time. The effective exposure time is therefore 1426 hours, or about 2 months of operation time.

A typical solar cell interconnector is made of silver and is about 25 microns thick. As the interconnector erodes, the resistance in the connection increases. This leads to power losses in the array. Assuming that losses become significant when about 10% of the interconnector's thickness has eroded results in an allowable erosion depth of 2.5 microns. This translates to an average erosion rate of no more than 1.2 microns/month on silver. Figures 6.28 and 6.29 show the cant and array angles at which such rates can be achieved on the baseline spacecraft configuration. In Figure 6.28, these regions are relatively small. Even at a cant angle of 45 degrees, array angles less than about 60° lead to unacceptable erosion rates on the interconnectors. Since the thruster does not need to be operated continuously, it should be possible to modify the thruster's duty cycle so it will only operate at these array angles. In fact, it may be possible to push the cant angle down to 40 degrees and still achieve a mean erosion rate of 1.2 microns/month by operating the thruster for short periods during each orbit.

It is important to note that the allowable cant and array angles depend strongly on the area of the array over which damage is allowed to occur. Figure 6.29 shows the erosion rate 30 cm. away from the point shown in Figure 6.28. The erosion rates here are quite a bit lower because the point is farther away from the thruster. The necessary mean erosion rate can clearly be achieved at 40 degrees and may be achievable at 35 degrees or less with the proper duty cycle. If a plume shield were also installed on the spacecraft, even lower cant angles may be achievable. Though the present work has not considered the effects of plume shields, the simulation has the capability to model very simple plume shields. This should be considered as a matter for future work.

Based on these results, a desirable range of operation for the thrusters would be at cant angles greater than 40 degrees and array angles greater than 50 degrees. These results only consider the effects of the plume on silver interconnectors and depend on several important assumptions. Since we assume that the entire array is biased at -96 V with respect to the thruster, portions of the array biased at more positive voltages should experience lower erosion rates. In addition, the interconnectors are typically mounted between cells and may be partially shielded from the plume by the edges of the cells. On the other hand, since the model does not consider the effects of non-normal impact angles, the effective erosion rate may be higher than calculated. We also neglect the relatively small amounts of erosion which occur at array angles greater than 90 degrees. Although *quasi3* includes models for the degradation of solar cell coverglasses, results from this model are not included in these results. As indicated in Chapter 4, the present model can significantly underpredict the erosions rates which will be experienced on operational arrays and further work is needed to improve the surface model to give accurate measurements of the degradation to solar cell coverglasses.

6.3 Summary

The axisymmetric PIC-DSMC model presented in Chapters 3 and 4 has been successfully extended to three dimensions and validated though comparison to experiment and to results shown in Chapter 4. The model has been applied to a realistic satellite configuration, and the results demonstrate the model's ability to evaluate thrusters placed in a variety of different orientations. The work shown in this chapter demonstrates the following points.

- The three dimensional PIC-DSMC model compares favorably to experimental data and almost exactly duplicates results from the axisymmetric simulation.
- The flow from the cathode of an SPT-100 thruster creates an axisymmetric neutral gas distribution, but has little effect on the structure of the plume itself.
- The PIC-DSMC model has been used to produce the first ever simulation of the interaction between the plume from a Hall Thruster and the surfaces of a communications satellite.
- The PIC-DSMC model has the ability to evaluate realistic spacecraft geometries with runs times of 12-20 hours on workstation class machines. This opens the possibility of using the model for spacecraft design work.
- A thruster's cant angle has a strong impact on the erosions rates experienced on critical surfaces like solar arrays. In practice, thrusters will require canting, so the effective specific impulse of the SPT-100 thruster will be lower than its laboratory value.
- The cant angle can also be used to limit the area of the array impacted by the plume.
- Lifetime erosion rates can be estimated from an assumed duty cycle. For the geometry studied in this chapter, cant angles of greater than 40 degrees and array angles of greater than 50 degrees appear to limit erosion damage to solar cell interconnectors. Lower cant angles may be possible with properly developed duty cycles that limit operation to time when the array angle is near 90 degrees or with the use of plume shields.

Overall, the results presented in this chapter verify the three dimensional PIC-DSMC plume model and demonstrate its ability to evaluate plume interactions on realistic spacecraft geometries.

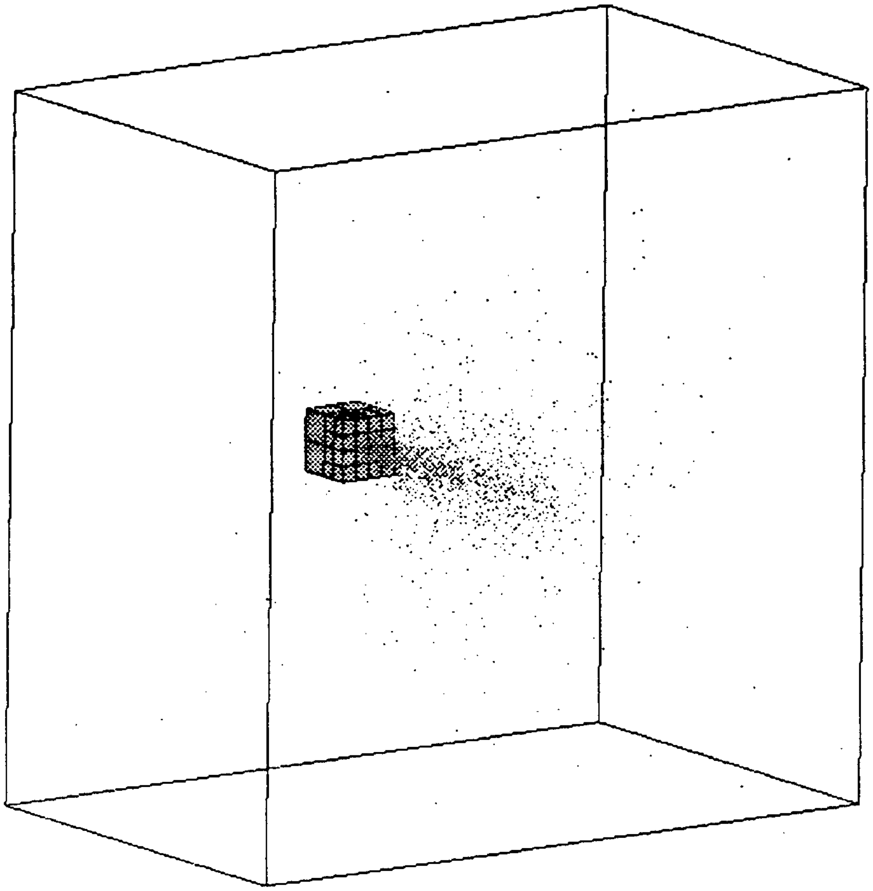


Figure 6.1: Simple Three Dimensional Simulation Geometry

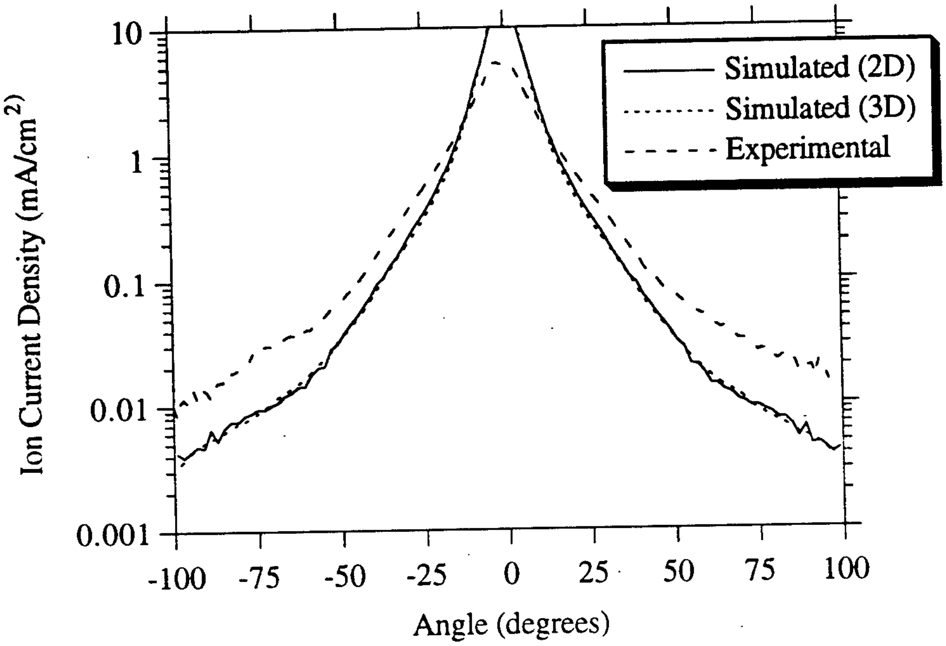


Figure 6.2: Experiment, 2D, and 3D Simulations of Ion Current Density
(z = 60 cm, P = 2.2 x 10^{-6} Torr)

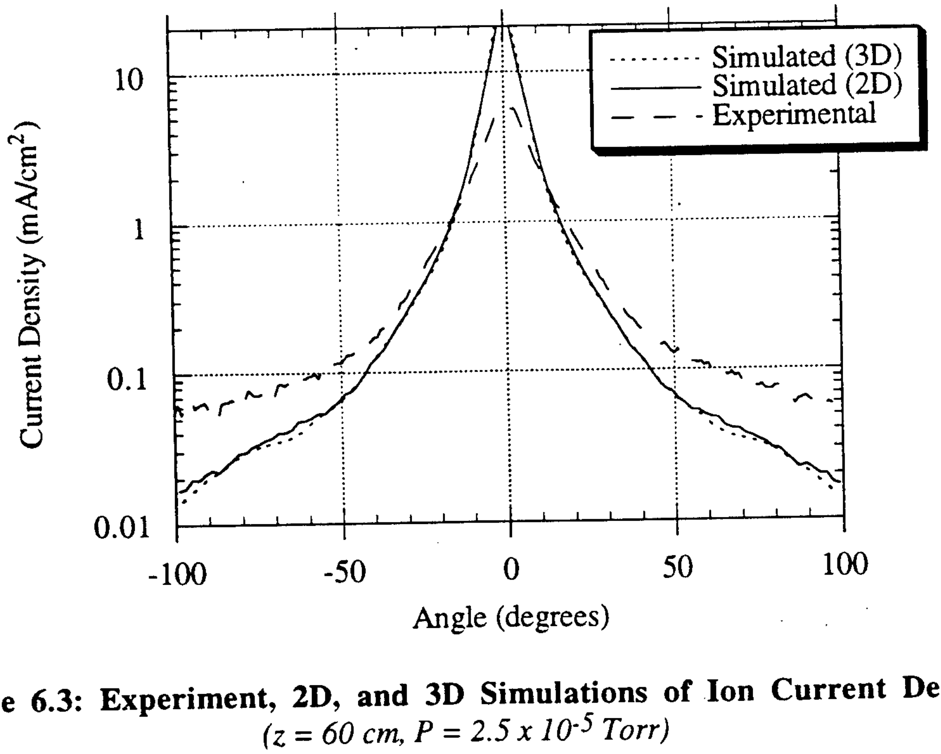


Figure 6.3: Experiment, 2D, and 3D Simulations of Ion Current Density
(z = 60 cm, P = 2.5 x 10^{-5} Torr)

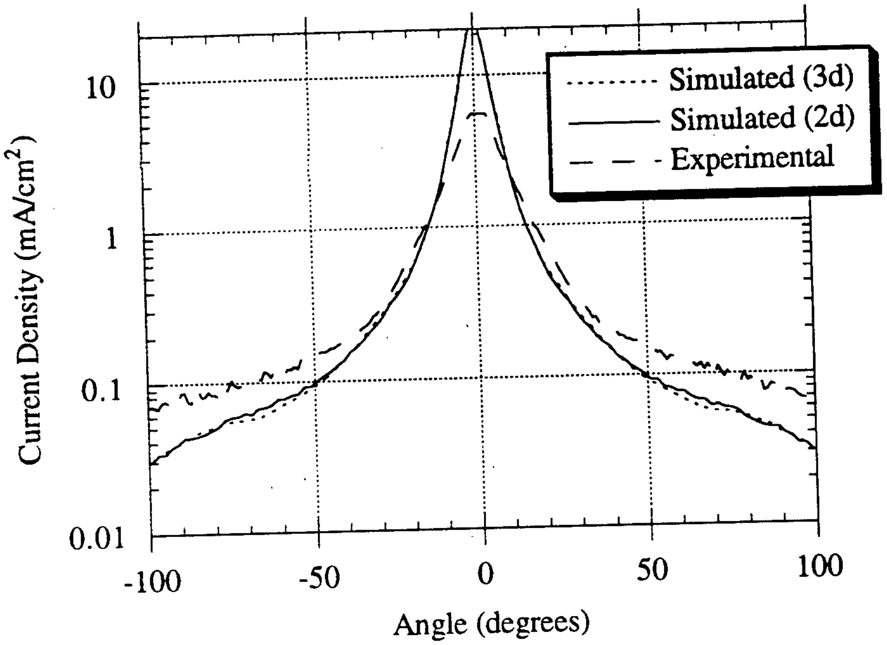
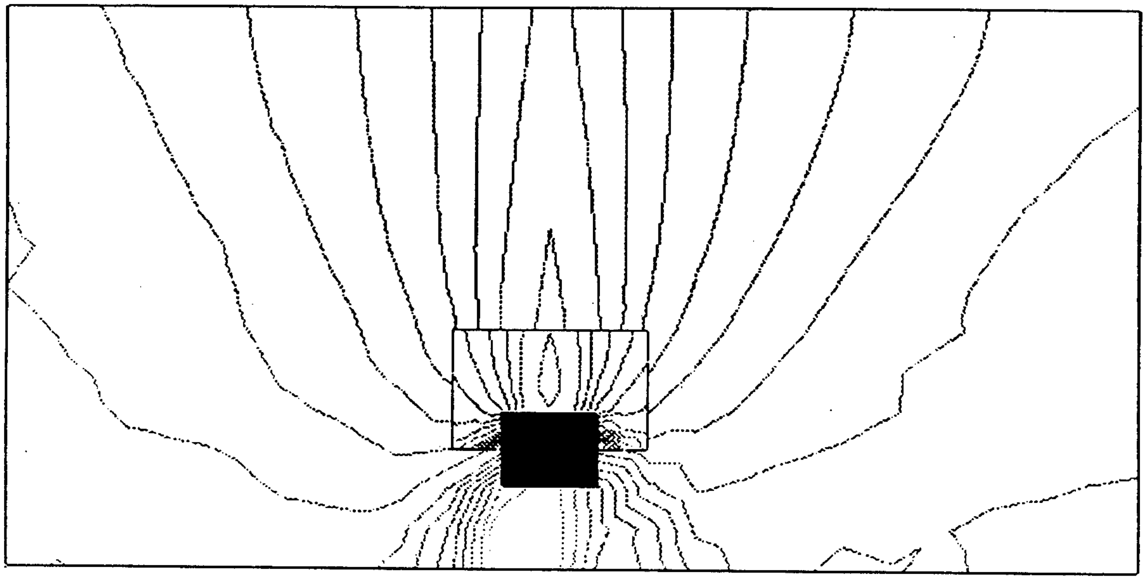
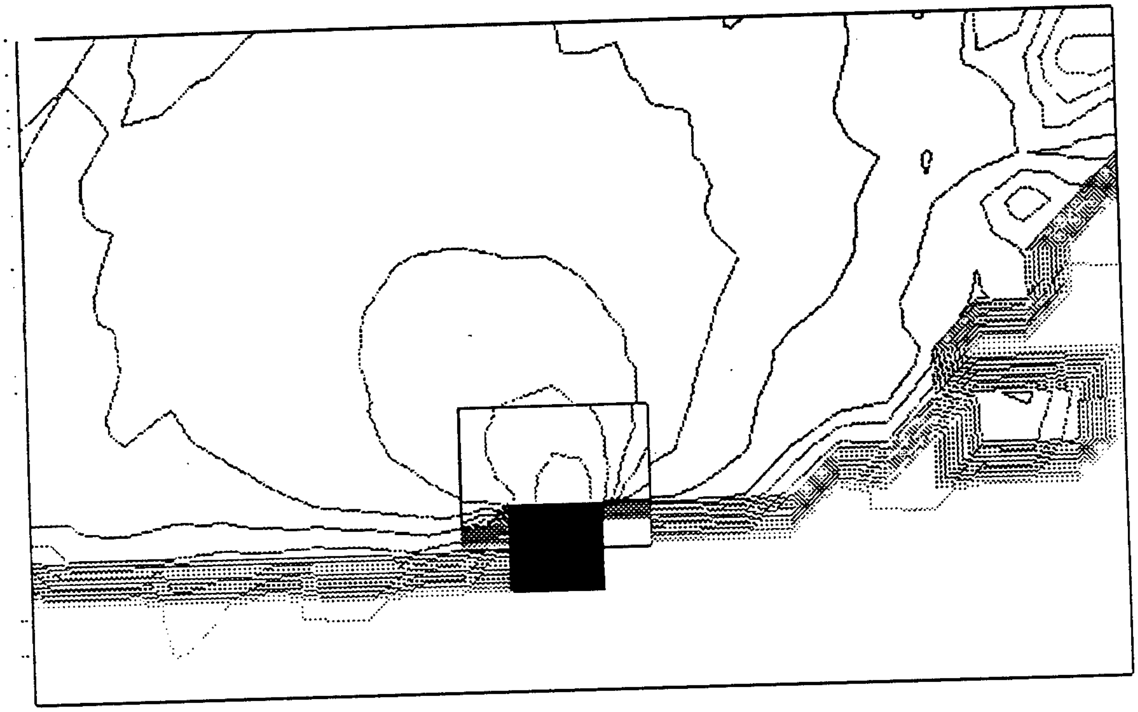


Figure 6.4: Experiment, 2D, and 3D Simulations of Ion Current Density
(z = 60 cm, P = 6.3 x 10^{-5} Torr)



Thruster is firing upwards. Solid box indicates thruster, hollow box indicates embedded grid. 20 contours are spaced logarithmically from 1 x 10^9 m^{-3} to 1 x 10^{18} m^{-3}

Figure 6.5: Contour Plot of Charge Density in SPT-100 Plume (3D)
(Pressure = 0 Torr)



Thruster is firing upwards. Solid box indicates thruster, hollow box indicates embedded grid.
20 contours are spaced logarithmically from 1 x 10^5 m^{-3} to 1 x 10^{18} m^{-3}
Irregular contours are due to low numbers of macro-particles in the backflow region.

Figure 6.6: Contour Plot of Neutral Density in SPT-100 Plume (3D)
(Pressure = 0 Torr)

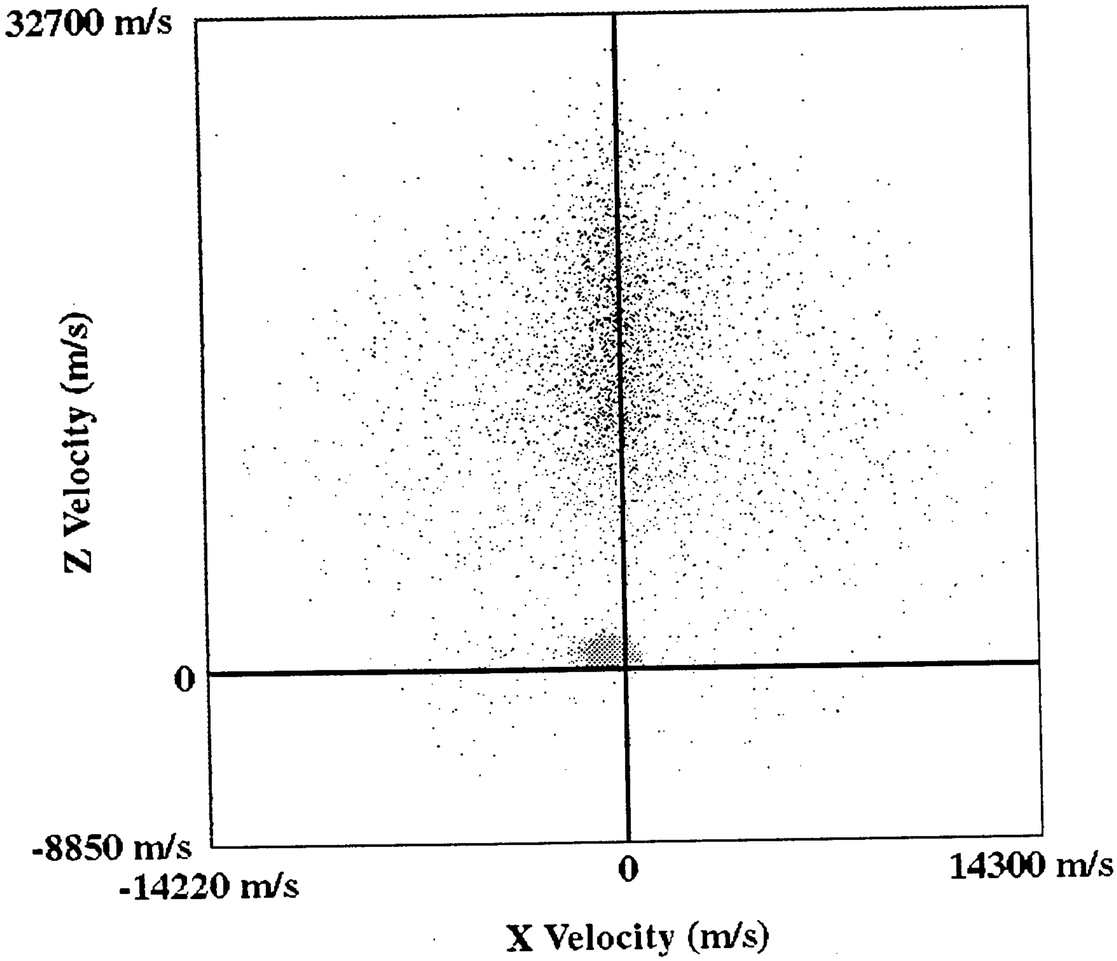


Figure 6.7: Phase Space Plot (X vs. Z Velocity) of SPT-100 Plume (3D)

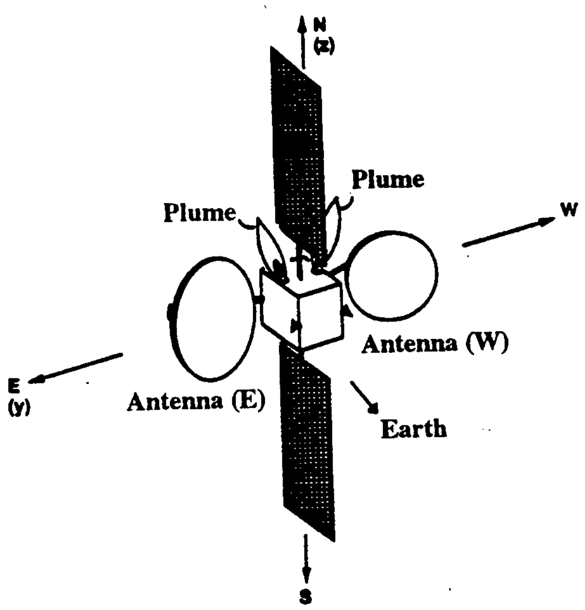


Figure 6.8: Thruster Orientation for North-South Stationkeeping
Adapted from Bogorad et al.[1992]

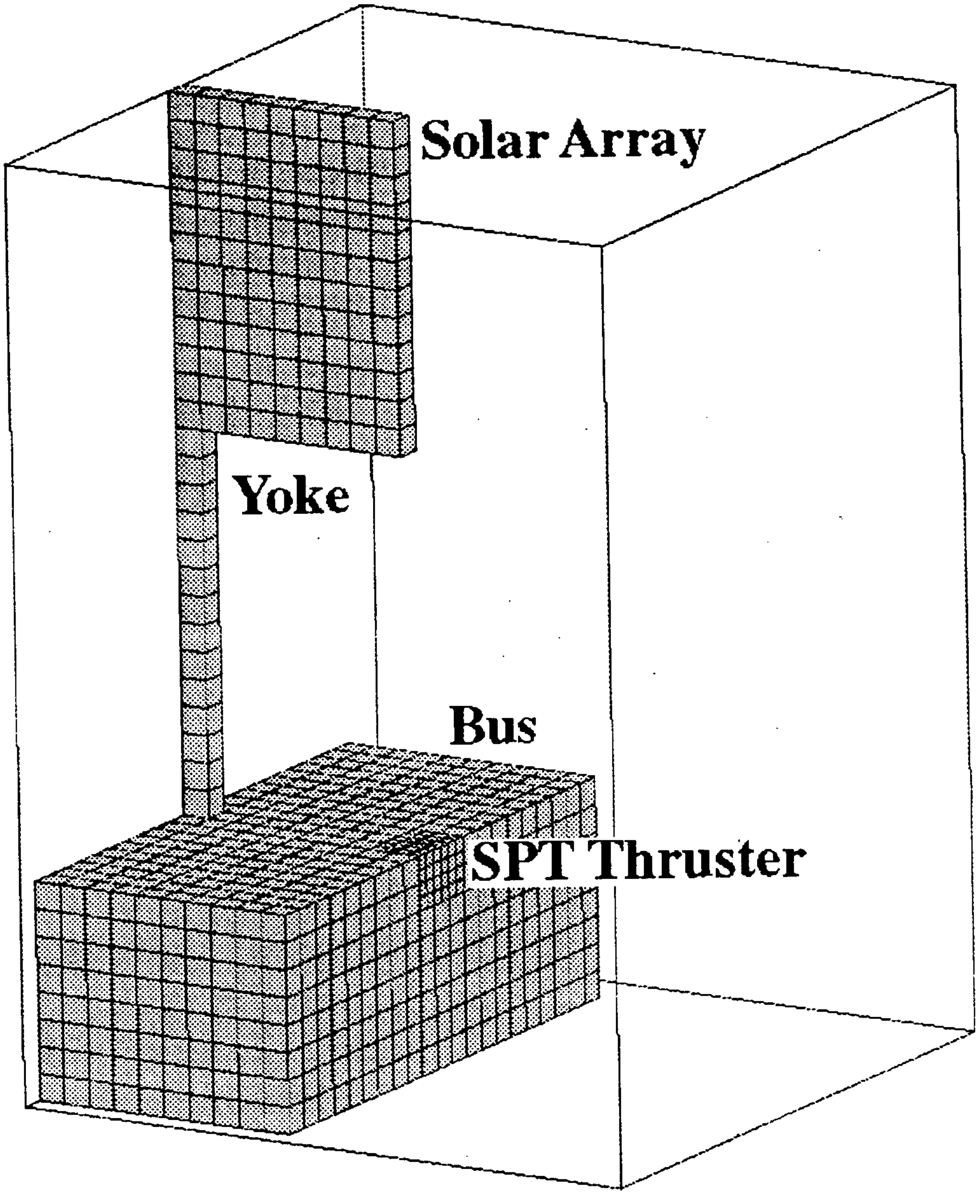


Figure 6.9: Geo-Comsat Satellite Configuration

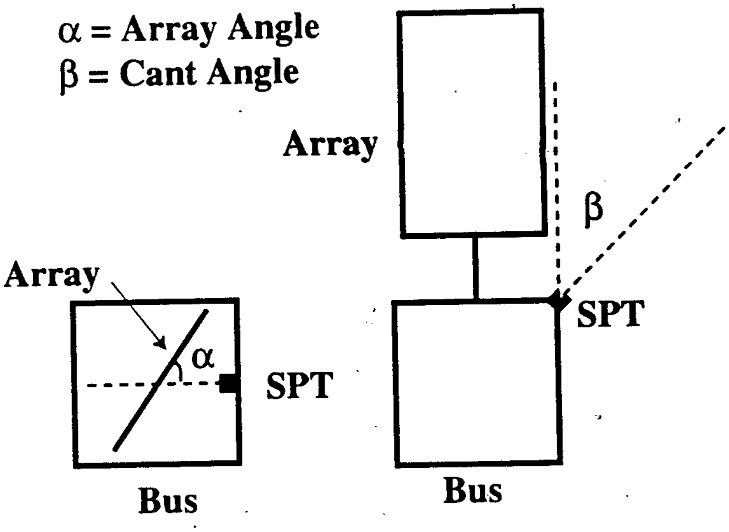


Figure 6.10: Definition of Cant Angle and Array Angle

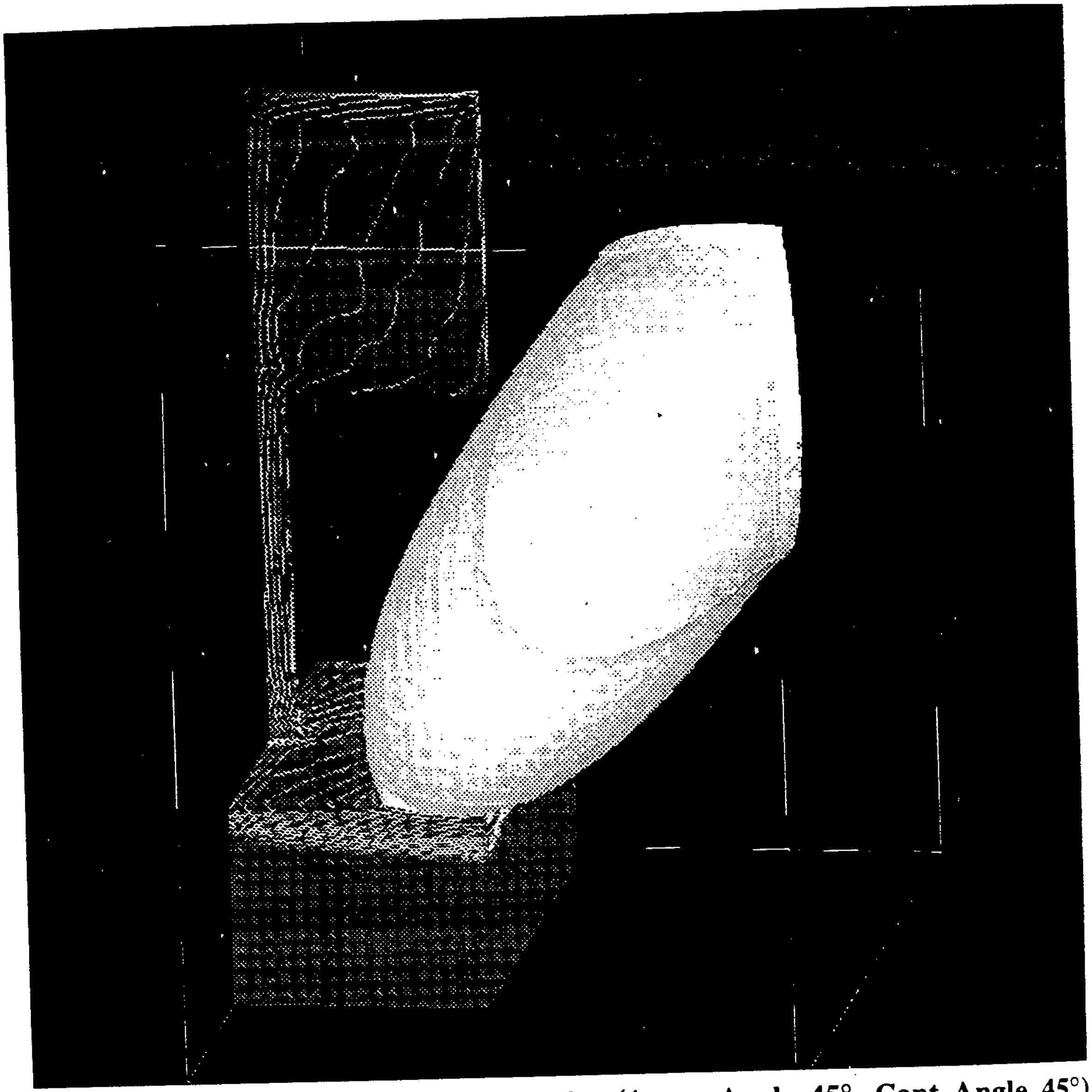


Figure 6.11: 3D Isopotential Surface Plot (Array Angle 45°, Cant Angle 45°)

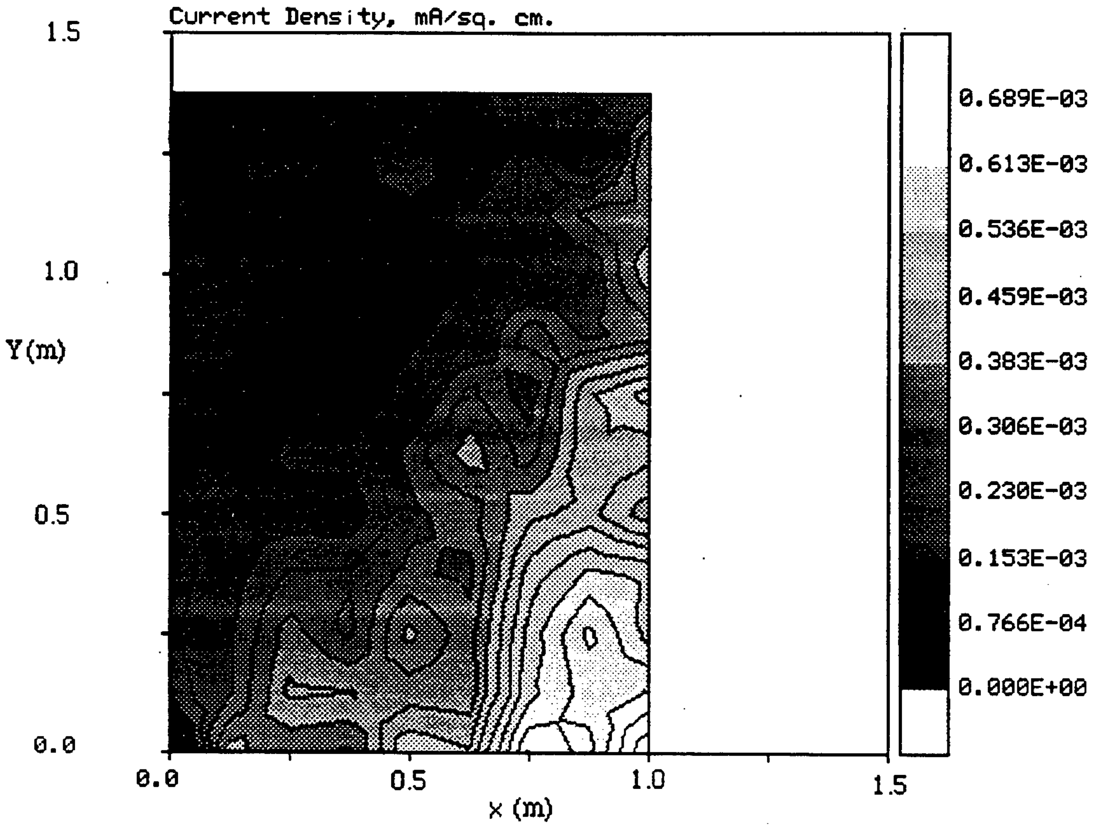


Figure 6.12: Ion Current Density on Solar Array (Array 45°, Cant 45°)

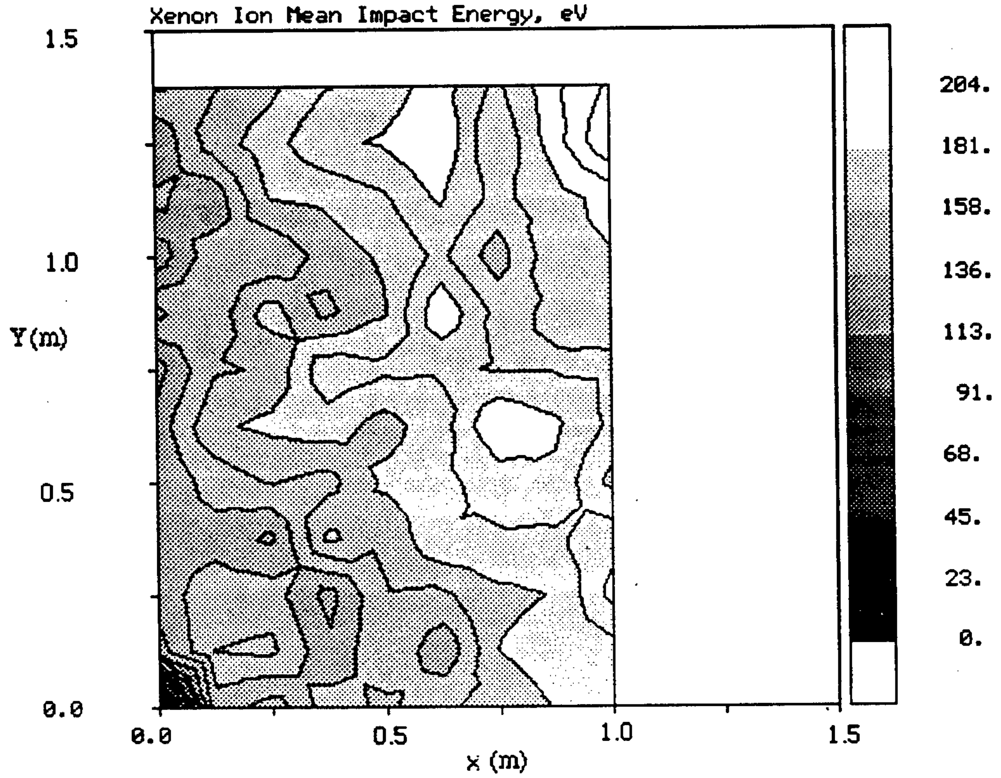


Figure 6.13: Mean Energy of Ions Striking Solar Array(Array 45°, Cant 45°)

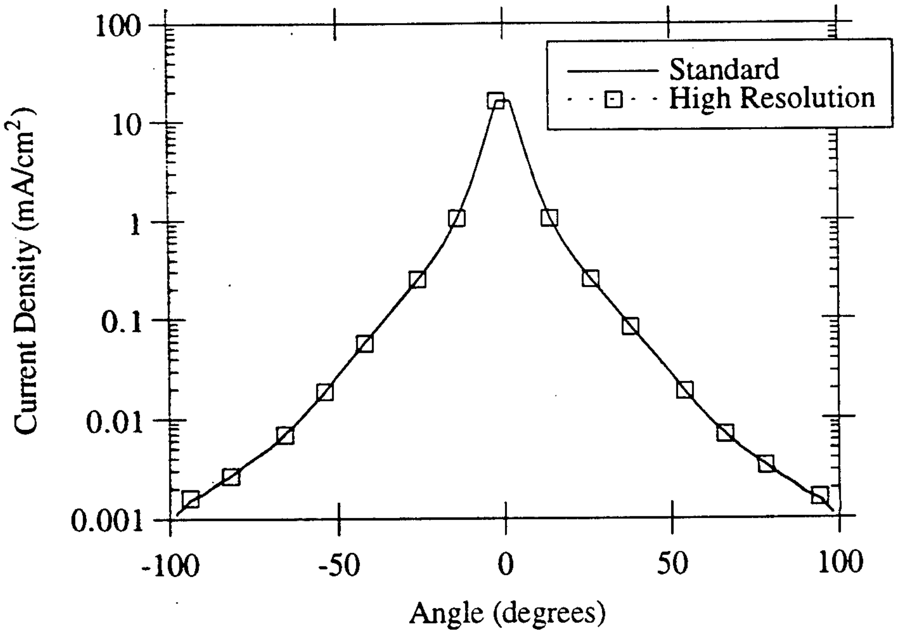


Figure 6.14: Ion Current Density Calculated w/ 420,000 and 1,870,000 Particles
(Standard = 420,000 Particles, High Resolution = 1.87 million Particles)

)

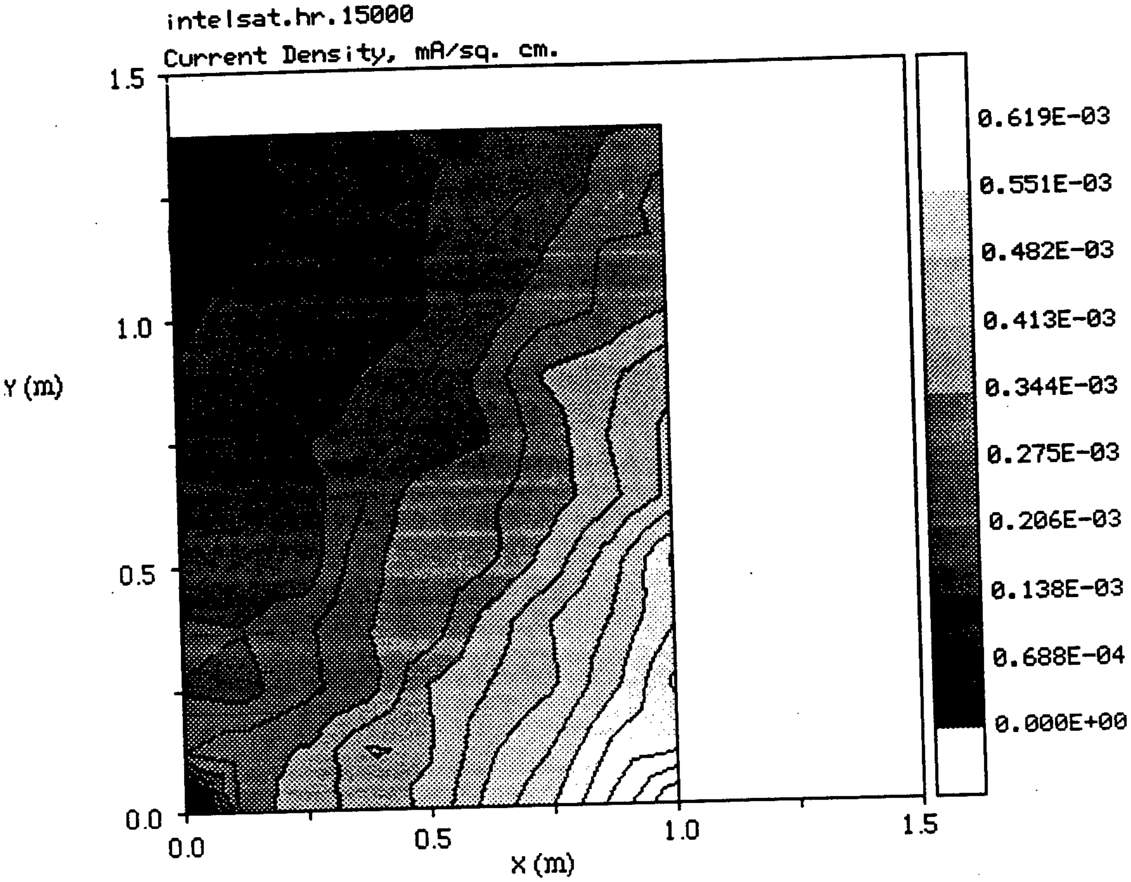


Figure 6.15: Ion Current Density on Array Calculated with 1,870,000 Particles
(Array 45°, Cant 45°)

icles

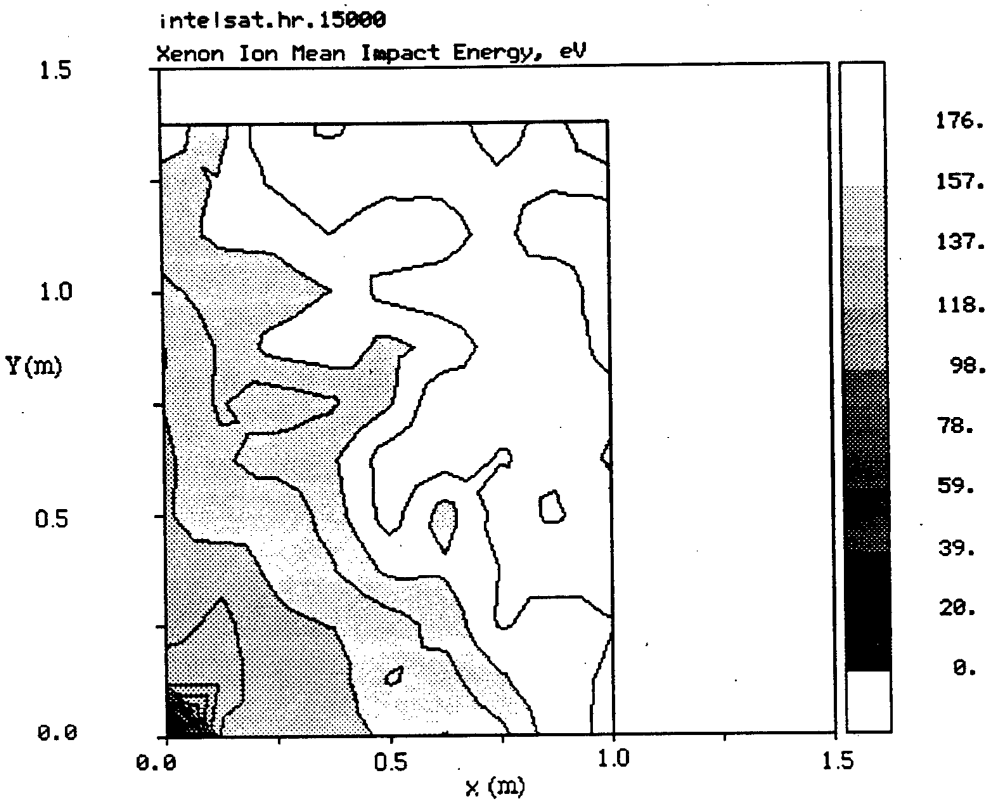


Figure 6.16: Mean Energy of Ions Striking Solar Array Calculated with 1,870,000 Particles
(Cant 45°, Array 45°)

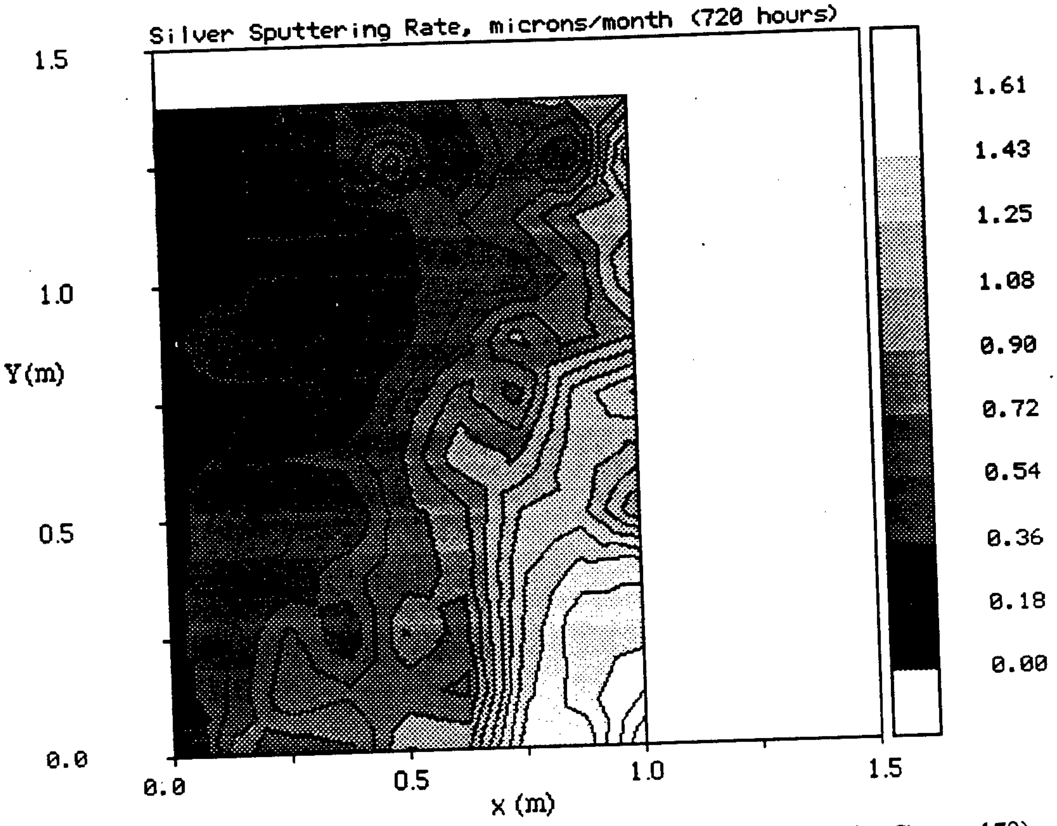


Figure 6.17: Erosion Rate for Silver (Array 45°, Cant 45°)

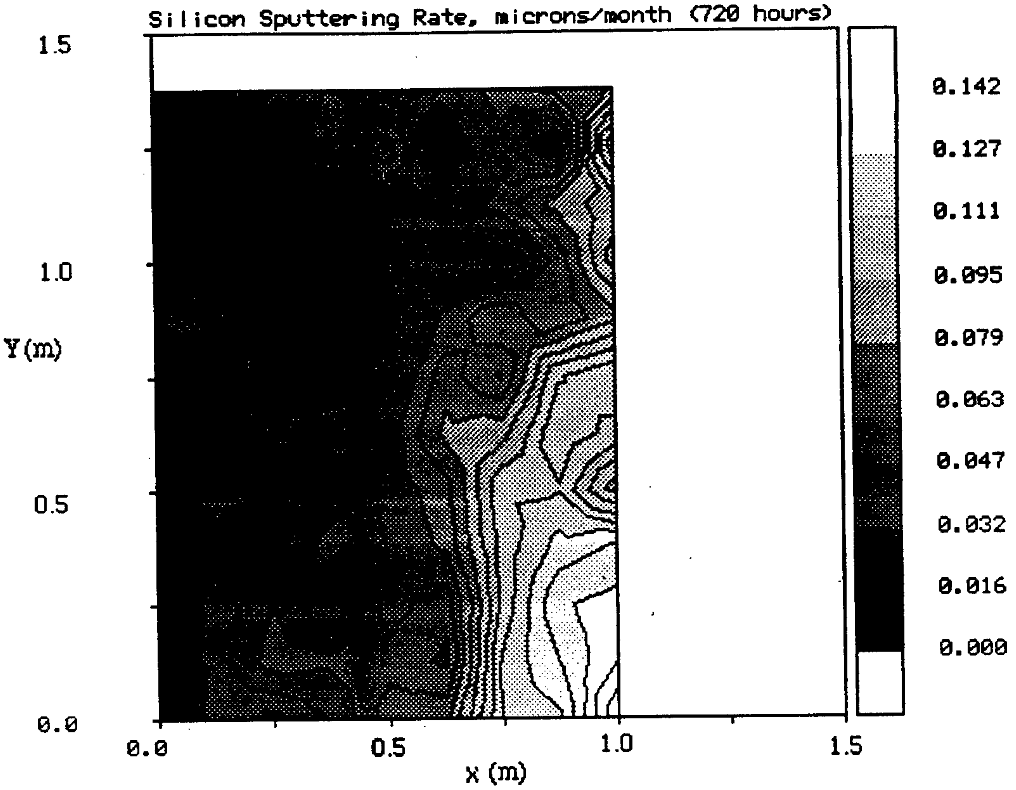


Figure 6.18: Erosion Rate for Silicon (Array 45°, Cant 45°)

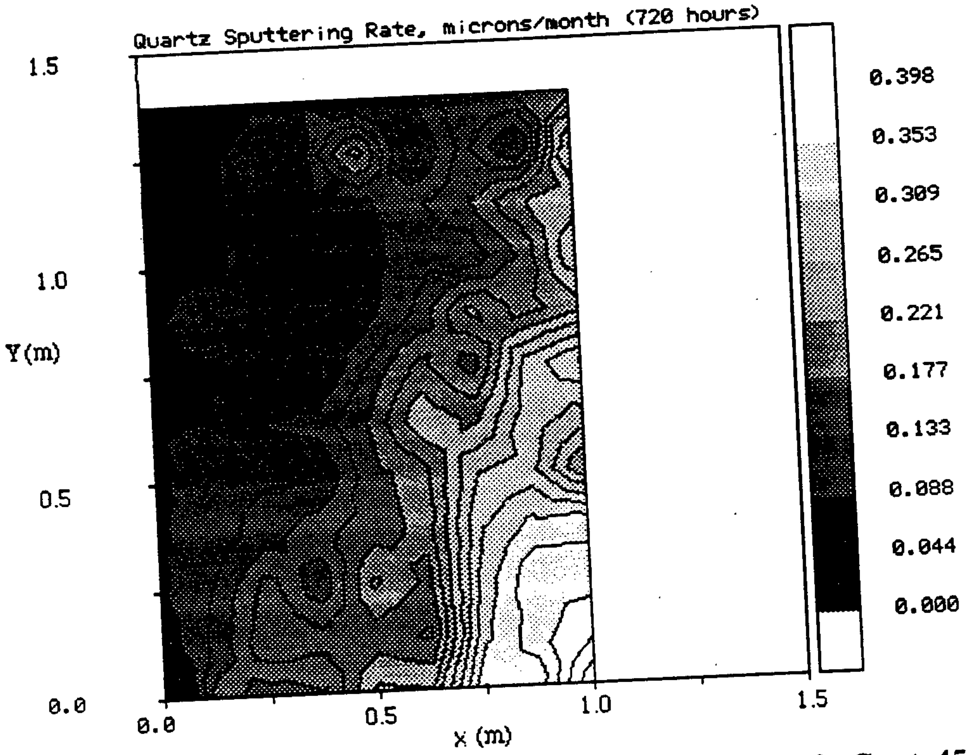


Figure 6.19: Erosion Rate for Quartz (Array 45°, Cant 45°)

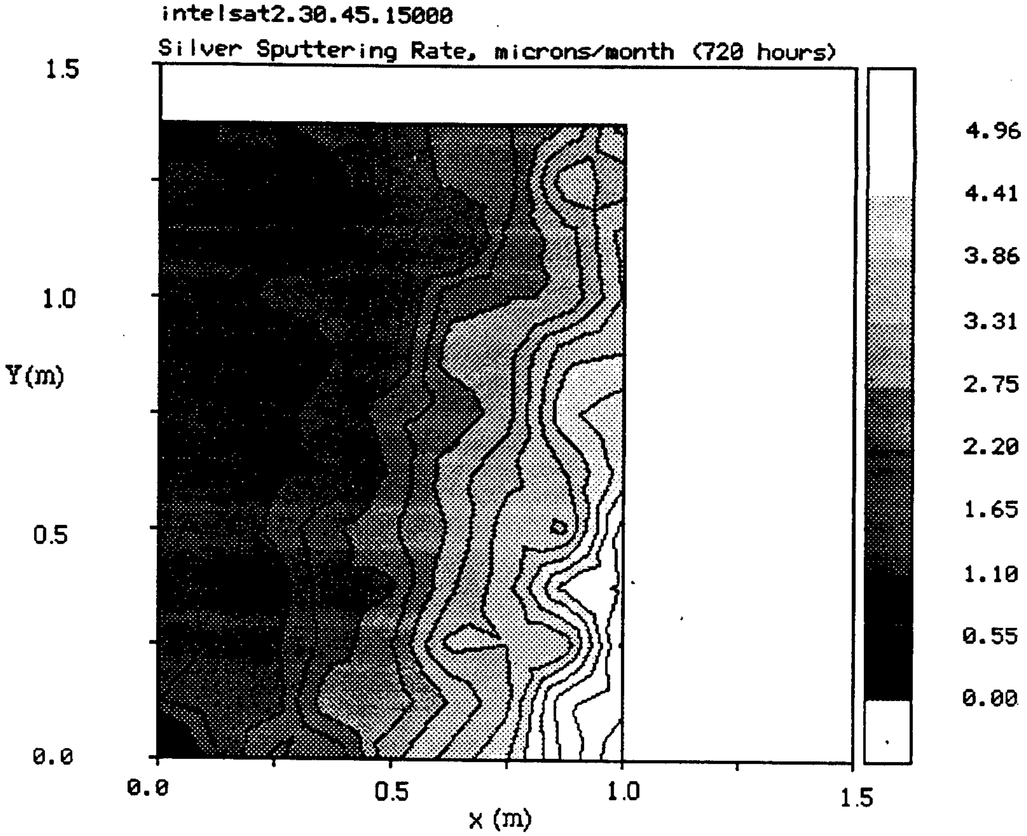


Figure 6.20: Erosion Rate for Silver (Array 45°, Cant 30°)

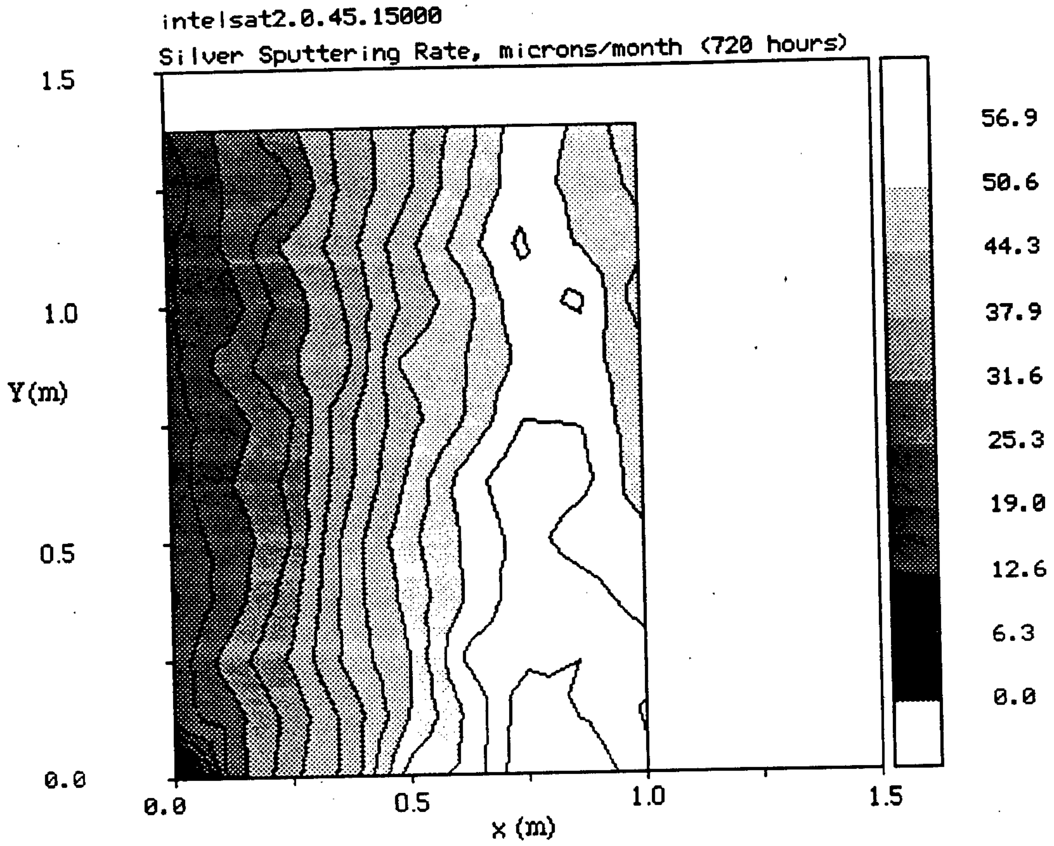


Figure 6.21: Erosion Rate for Silver (Array 45°, Cant 0°)

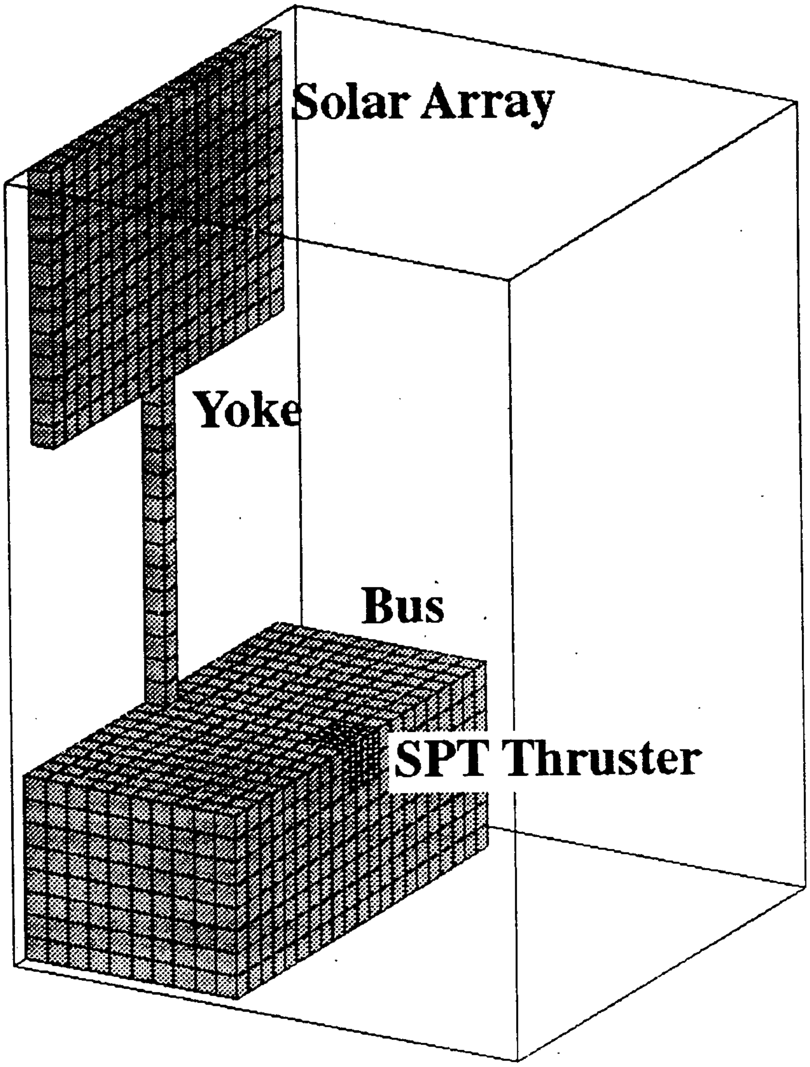


Figure 6.22: 3D Simulation Geometry for Array Angles Greater than 45 Degrees

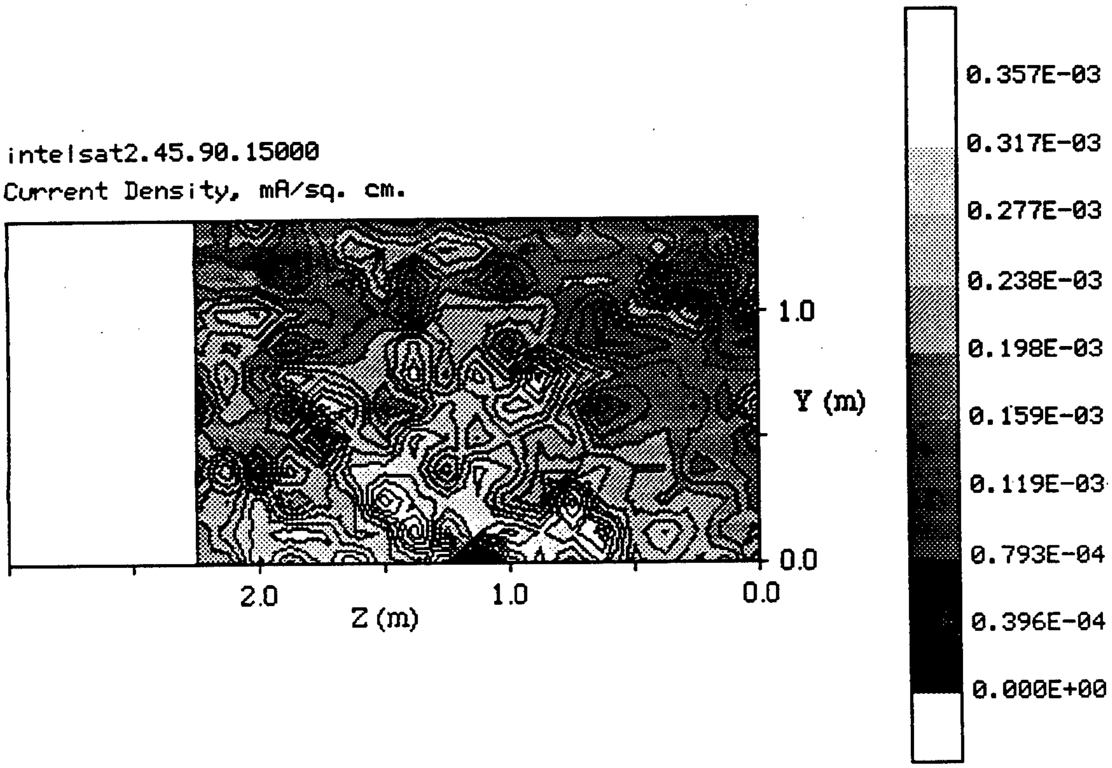


Figure 6.23: Current Density (Cant 45° degrees, Array 90°)

rees

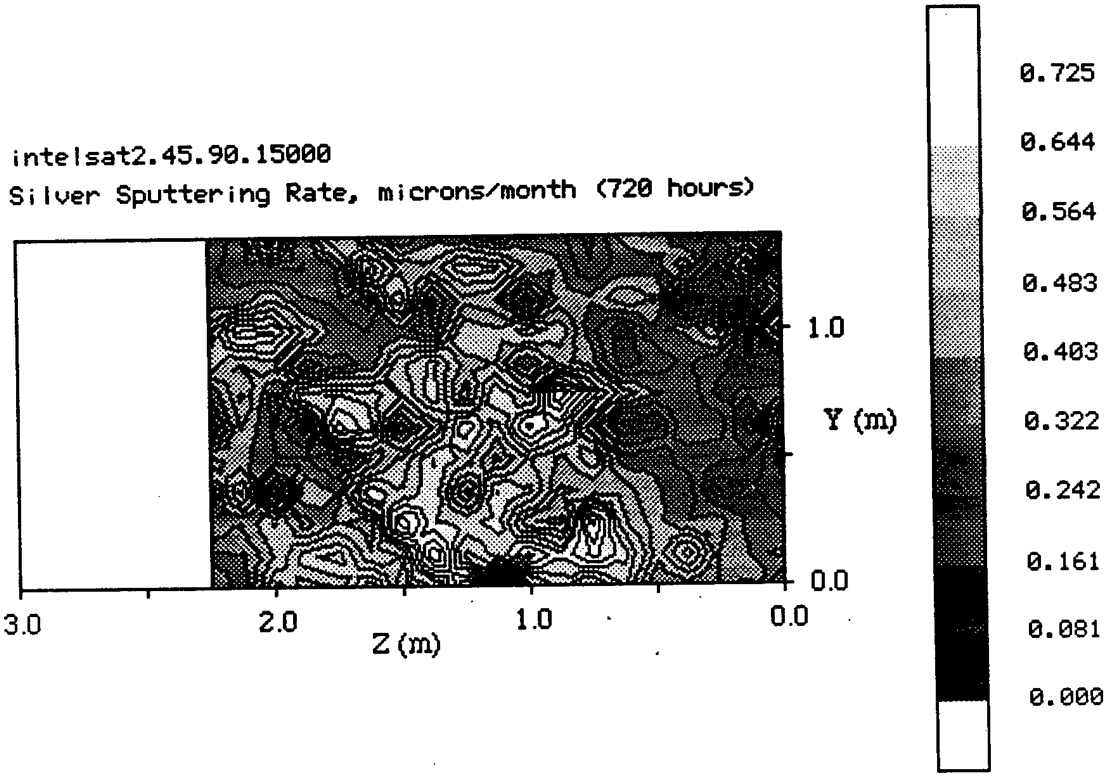


Figure 6.24: Erosion Rate for Silver (Cant 45°, Array = 90°)

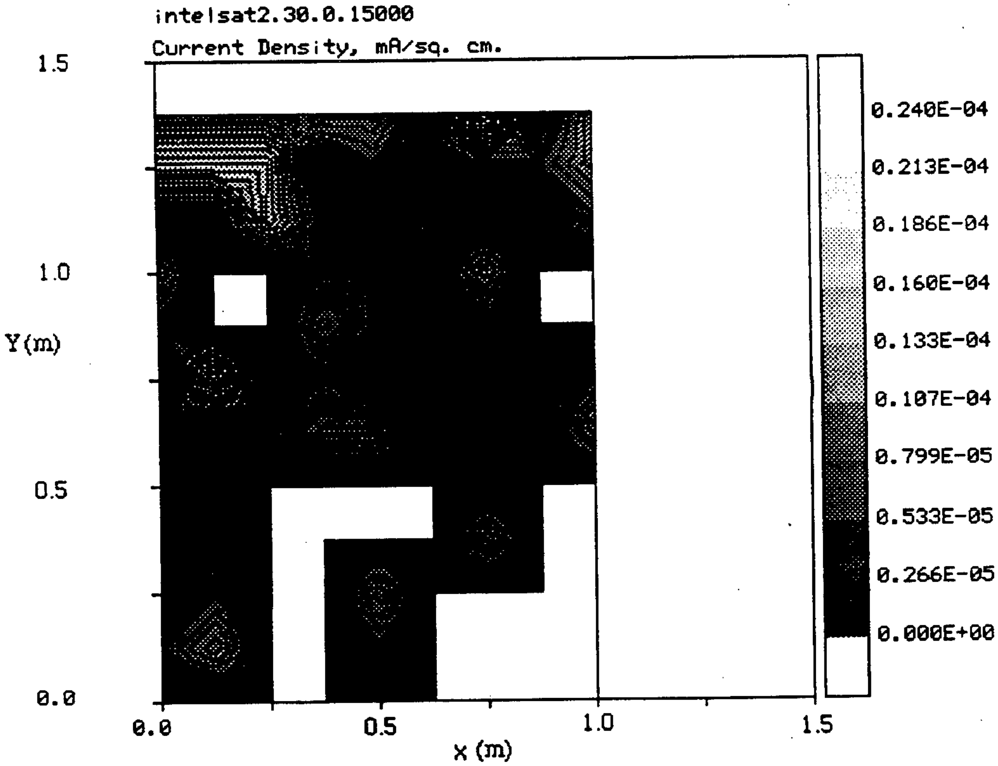


Figure 6.25: Ion Current Density on Solar Array (Array 0°, Cant 30°)
(Plot is very noise because current shown is near to simulation's resolution limit)

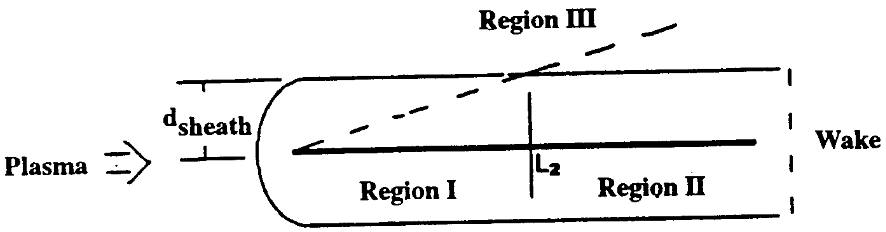


Figure 6.26: Plasma Flow Field Around a Plate a Zero Angle of Attack

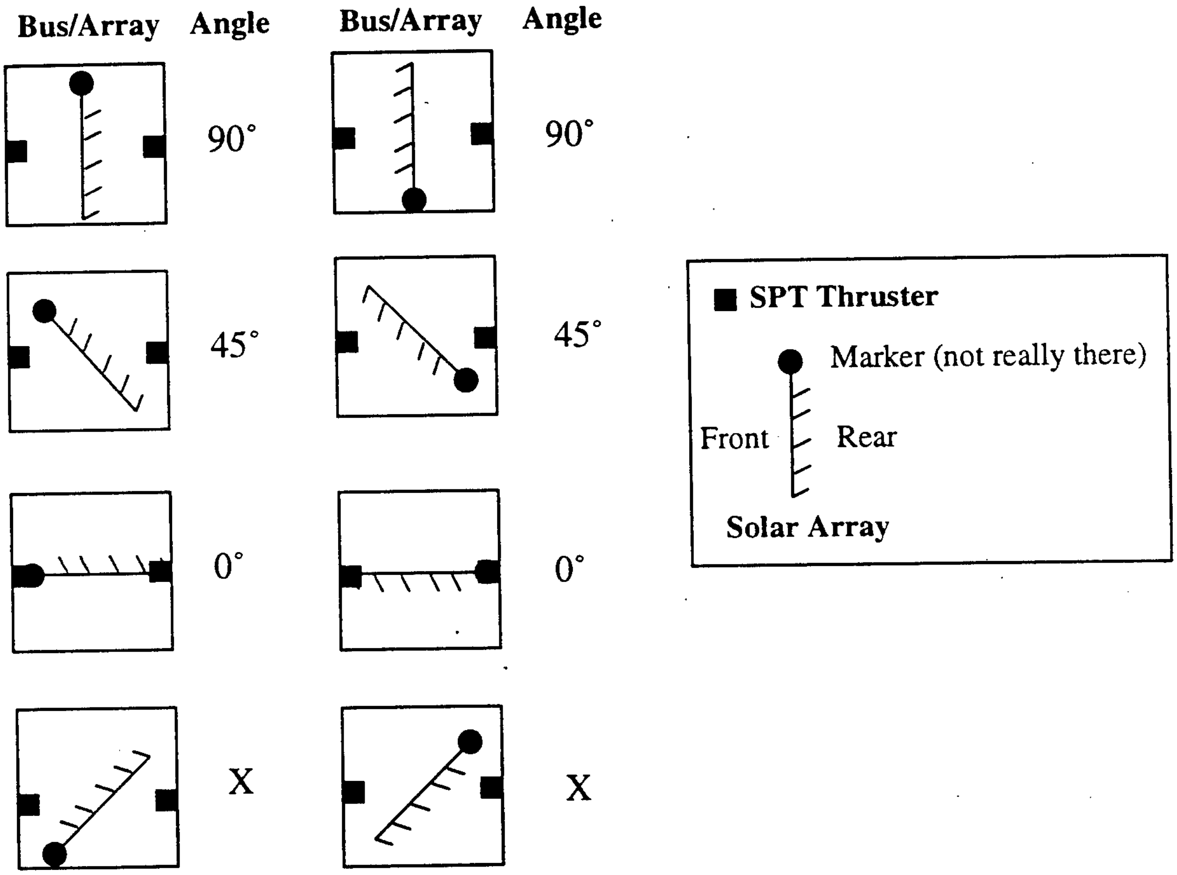


Figure 6.27: Solar Array Orientation and Array Angle

Erosion Rates in microns/month

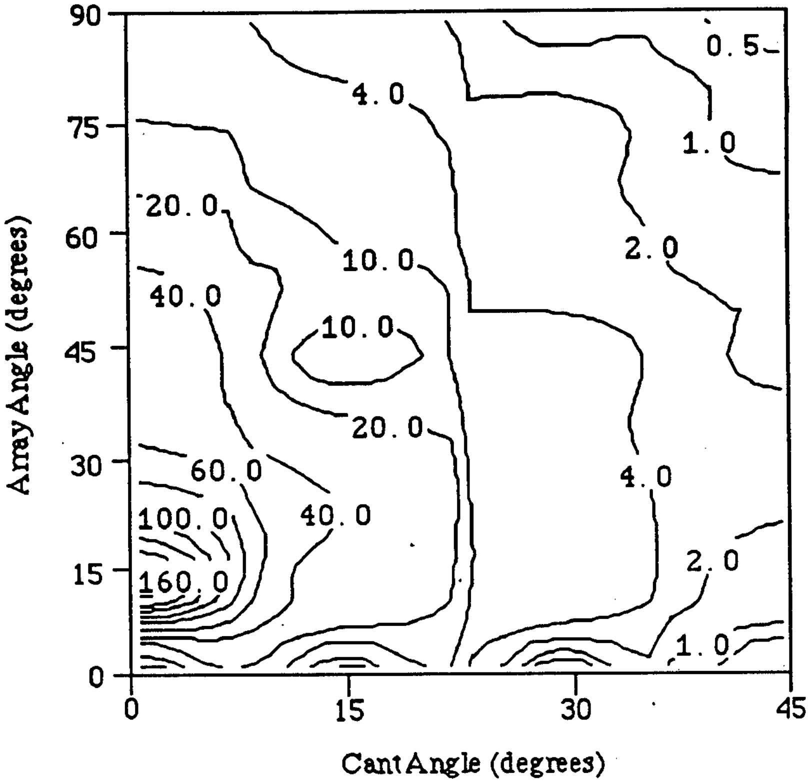


Figure 6.28: Erosion Rate for Silver at Corner of Array
SPT-100 Thruster, Mass Flow = 5.37 mg/sec., I_{sp} = 1610 sec.,
Yoke Length = 1.9 m, Array Width = 1.14 m
(Contours extrapolated from at total of 21 points, see Table 6.1)

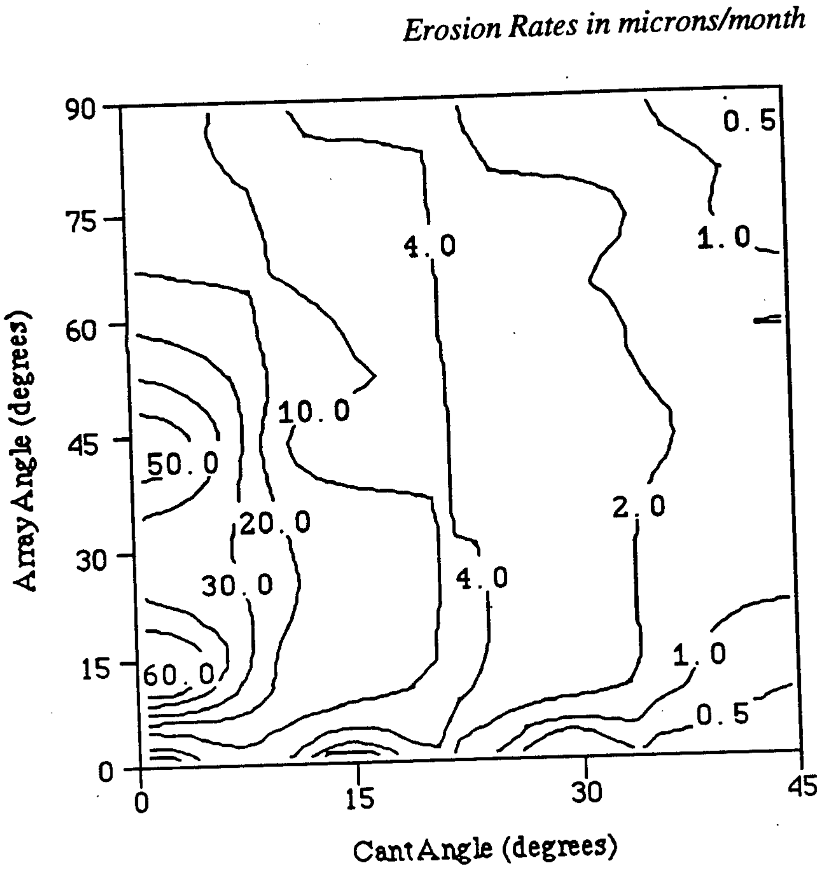


Figure 6.29: Erosion Rate for Silver 0.3 m from Corner of Array
SPT-100 Thruster, Mass Flow = 5.37 mg/sec., I_{sp} = 1610 sec.,
Yoke Length = 1.9 m, Array Width = 1.14 m
(Contours extrapolated from at total of 21 points, see Table 6.1)

Chapter 7 Conclusions

A computational model has been constructed of the expanding plasma plume from an SPT thruster. The model is based on theoretical work suggesting that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. A quasi-neutral Particle in Cell-Direct Simulation Monte Carlo (PIC-DSMC) plasma model is used as the basis for the simulation. This work represents, to the best of our knowledge, the first use of a combined PIC-DSMC model in any application. Both axisymmetric and three dimensional models have been developed, partially validated against experimental data, and shown to model the plume from a Hall Thruster accurately on meter length scales. The methods used in this work can also be applied to the plumes from some other EP devices such as ion thrusters and anode layer thrusters (though the full DSMC model may be overkill for the plume from an ion thruster). The next two sections summarize and review the various findings made in this thesis. They describe specific observations made of the plumes from Hall Thrusters and outline the contributions which this thesis makes to the state of the art in the computational modeling of plasmas and in the understanding of spacecraft-plume interaction issues.

7.1 Summary of Results

A quasi-neutral PIC-DSMC model of an expanding plasma plume has been developed and implemented for axisymmetric and three-dimensional geometries. The simulation is the first comprehensive model of the plume from a Hall Thruster and its interaction with the surfaces of a spacecraft. The axisymmetric version of the PIC-DSMC model has been used to verify the PIC-

DSMC method and to simulate an SPT-100 operating under conditions that can not be duplicated in ground experiments. Based on comparisons to multiple sources of experimental data, the following conclusions have been reached.

- The PIC-DSMC model accurately simulates the large scale structure of the plume. The model has excellent agreement with ion current density measurements taken by Marrese/Gallimore at the University of Michigan and fair agreement with similar data taken by Manzella at NASA Lewis.
- The PIC-DSMC model is currently unable to model details of the ion distribution function and match the ion current density distribution at the same time. This limitation is not due to physical problems in the simulation but is due to assumptions made in the plasma source model. In particular, a higher ion temperature is used in the present source model than the measured experimental value. Some aspects of the experimental data are inconsistent with current understanding of the SPT-100 thruster, and detailed theoretical work is needed to determine the characteristics of the plasma emerging from the thruster.
- The quality of the results depends on the quality of the SPT-100 source model. The source model itself is based on the best data available at this time. However, the original data is old and the quality of the measurements themselves is unclear. Better measurements of the ion current density in the near field region are needed to improve the source model. Alternately, better theoretical models of the thruster itself could be used to improve the source model.
- Some existing experimental data may need to be examined closely. Beam current have been reported which exceed the actual discharge current and some RPA data appears to give I_{SP}'s higher than actually observed in SPT thrusters.
- The surface interaction model accurately models the erosion of silver. It is less accurate when applied to solar cell coverglasses because the sputtering coefficients of coverglasses have not been reported in the general scientific literature.

Overall, comparisons with experimental data serve to partially verify the basic PIC-DSMC method when applied to plumes from Hall Thrusters. The weakness with the existing method is that the results are sensitive to the quality of the SPT-100 plasma source model. This model is based on experimental data taken in the early 1980's and its accuracy is limited by the quality of the data itself. Given better data, it may be possible to construct better source models that give better agreement with RPA measurements. The sensitivity of the results to the source model has not been quantified. Future work should be conducted to improve the source model and determine the sensitivity of the simulation to parameters in the source model.

One advantage of the PIC-DSMC method is that it can be used to make detailed studies of the structure of the plume region. Based on such studies, the following conclusions have been drawn.

- When an SPT-100 thruster operates in vacuum, the majority of the CEX ions are created within three thruster widths of the channel exit.

- The CEX ions form a wing structure to the side and rear of the plume. However, the main beam has a high divergence angle. As a result, the CEX wing merges with the main beam and is not visible in measurements of the ion current density.
- Increasing the neutral flow in a plasma thruster not only results in more backflow current, but also changes the fundamental character of the plume region. As the backflow increases, the CEX ions form distinct wing structures which affect the potential structure in the plume region.

Finally, the axisymmetric model has been used to simulate an SPT-100 thruster operating in a true vacuum. This matches conditions which will be seen by thrusters operating on GEO comsats and can not be duplicated in ground based tests. The following conclusions are reached.

- Ground tests may significantly overpredict the backflow current from SPT thrusters, though tests conducted at very low pressures (~2 x 10^{-6} Torr) may accurately reflect operational conditions.
- The erosion rate drops by approximately an order of magnitude with each 20 degrees from the thruster centerline.
- The erosion rate drops with the square of the distance from the thruster exit.

A three dimensional plume model has also been constructed based on the quasi-neutral PIC-DSMC method. The model has been verified through comparison to the axisymmetric simulation and has been used to evaluate the erosion damage on a typical GEO comsat configuration. The results of the simulation can be summarized as follows.

- The PIC-DSMC model has been used to produce the first ever simulation of the interaction between the plume from a Hall Thruster and the surfaces of a communications satellite.
- The PIC-DSMC model has the ability to evaluate realistic spacecraft geometries with runs times of 15-20 hours on workstation class machines. This opens the possibility of using the model for spacecraft design work.
- A thruster's cant angle has a strong impact on the erosions rates experienced on critical surfaces like solar arrays.
- Lifetime erosion rates can be effectively estimated by assuming a thruster duty cycle. A typical geo-comsat configuration was studied and cant angles of greater than 40 degrees and array angles of greater than 50 degrees were found to adequately limit erosion damage to solar cell interconnectors. Lower cant angles may be possible with properly developed duty cycles that limit operation to time when the array angle is near 90 degrees or with the use of plume shields. No judgment was made with respect to coverglass erosion.

These results represent the state of the art in Hall Thruster plume simulations. This work is the only three dimensional Hall Thruster plume study made to date and is the first published effort to model the effect that the plume has on the surfaces of a satellite. Overall, our results show that the erosion damage caused by Hall Thrusters is manageable and should not represent a barrier to their use on commercial and military spacecraft. However, the thrusters will either have to be canted away from sensitive surfaces, thus lowering their effective I_{sp}, or plume shields will have to be used, thus imposing a mass penalty on spacecraft using these devices for propulsion.

7.2 Contributions to the State of the Art

The work presented in this thesis is a significant contribution to the modeling of spacecraft-plume interactions and to the numerical simulation of plasmas. This work represents the first time that the PIC and DSMC algorithms have been combined to produce a particle model of a plasma plume which includes both long range Coulomb interactions and a fully developed collision model and used to model problems for the EP community. To the best of our knowledge, this work represents the first use of a combined PIC-DSMC model in any application. Most existing particle models for collisional plasmas are based on the PIC-MCC method which do not explicitly track both collision partners. By modeling and tracking the effects of collisions on all particles, the PIC-DSMC method extends the regimes of plasmas which can be effectively modeled using particle methods. The PIC-DSMC method not only covers the regime of plasmas covered by conventional PIC-MCC methods but also include plasmas in which neutral-neutral collisions are a factor and/or in which collisions effect the underlying distribution of all species in the plasma. This allows one to model the plasmas emitted by Hall thrusters in the course of their operation and to model hypothetical thrusters with higher neutral flow rates. The main limitation on the PIC-DSMC method is that it can not presently be used to simulate short range coulomb collisions. This limits the range of ion densities and temperatures which can be modeled using this method.

In addition, the model presented in this thesis uses the assumption of quasi-neutrality to greatly enhance the speed of the PIC simulation. While conventional PIC methods solve for the electrical potential by inverting Poisson's equation, we assume that the plasma is quasi-neutral and determine the electrical potential by inverting Boltzmann's relationship. This assumption results in a plume model which requires significantly fewer calculations than any previous numerical model of the plumes emitted by Hall or Ion thrusters. Previous models of three dimensional plume-spacecraft interactions required days of computing time on massively parallel machines. The 3D quasi-neutral PIC-DSMC model used in this work simulates equivalent domains with only 24 hours of computing time on a workstation class machine. This represents a significant advance in the speed of numerical models of plasma plumes and, for the first time, opens the possibility of using a particle based plume simulation as a spacecraft engineering design tool. The quasi-neutral assumption precludes the use of the simulation in some regimes, most notably in wake regions where the Debye length is of the same order as the body length scale. However, as demonstrated in this work, the quasi-neutral PIC-DSMC method is able to model the dense regions of a plasma plume which are of greatest interest to spacecraft designers. Since the methods itself is not application specific, the quasi-neutral PIC-DSMC method can also be used to simulate plasmas for non-space applications such as semiconductor processing. Overall, the computational methods presented in this paper extend the regime of plasmas which can be simulated using particle methods and greatly increase the speed of particle based plasma computations. This represents a significant advance in plasma simulation algorithms.

This work also represents a significant advance in the simulation of spacecraft-plume interactions, particularly with regard to Hall Thrusters. Although a great deal of experimental work has been carried out on Hall Thrusters, relatively little analytical work has been carried out on the plumes produced by these devices. The few models which do exist are empirical in nature and based on fits to experimental data. This work represents the first comprehensive numerical model of the plume from a Hall Thruster. Unlike other models, our simulation models the fundamental physics of the plume region including CEX collisions and collective Coulomb interactions. This work is also the first fully three dimensional model of the plume from a Hall Thruster and is the first published effort to analyze the interaction between a Hall Thruster and a realistic spacecraft configuration. Because the methods used in this work are valid for many quasi-neutral plasmas, the simulation can also be used to model other EP devices such as TAL and ion thrusters. In summary, this work provides designers with a prototype computational tool that can predict the interaction between the surfaces of a satellite and the plasma plumes produced by EP devices. As the first comprehensive model of the plume from a Hall thruster and the first three dimensional plasma plume-surface interaction model which can run on workstation class machines, it represents a significant advance in the modeling of plume-spacecraft interactions.

7.3 Recommendations for Future Research

This work raises a number of issues with respect to Hall Thrusters and there are many improvements which can and should be made to improve the existing simulation. The following issues have been identified as areas in which improvements should be made to the existing model.

- Analysis of experimental data indicates that the SPT-100 thruster may be emitting high energy ions at relatively high divergence angles. These ions may have a significant impact on exposed surfaces. The mechanism by which these ions are created needs to be studied and understood and the ion themselves need to be included in the plasma source model.
- The empirical source model should be improved to increase the level of agreement between the simulation and experimental measurements. The present model consistently overpredicts currents along the centerline and requires the use of an axial ion temperature that is inconsistent with existing experimental data. It should be possible to increase the overall accuracy of the simulation by modifying this source model. Possible approaches include taking better measurements of the current in the near field region or incorporating a theoretical model of the thruster itself into the plume model.
- One weakness of the existing model is that it depends on an imperfect plasma source model. An effort might be made to quantify the sensitivity of the plume simulation to the variations in the source model. This could be accomplished by defining plume parameters as functions of source parameters. Such a function would have to be defined empirically and

would be highly non-linear. However, if such a function could be defined, it could be linearized and used to quantify the sensitivity of the plume to small variations in the source model.

- If computational resources become available, further efforts should be made to verify the quasi-neutral PIC-DSMC method against theory. A supercomputer, for instance, could be used to run non-quasi-neutral PIC-DSMC simulations. These results could be compared to results from the quasi-neutral PIC-DSMC simulation to see if the analytical source model correctly calculates sheath potentials and to confirm that the wake region has no influence on the overall accuracy of the simulation.
- Improved measurements should be taken of sputtering coefficients for Xenon and used to improve the surface-interaction model. The major limitation on the accuracy of the surface sputtering model is the lack of data on off-normal sputtering coefficients for Xenon and the lack of accurate sputtering measurements for solar cell coverglasses. Experimental efforts in this area could greatly enhance the accuracy of the surface interaction model.
- An unstructured mesh implementation should be seriously considered. Though an unstructured mesh would slow the simulation and add considerable complexity, it would allow one to model complex and arbitrary spacecraft geometries including irregular surfaces like antennae.
- A model for the refraction/propagation of EM waves should be added to the simulation, perhaps in a post processing mode. Some efforts have already been undertaken to model the impact of the plume on radio transmissions [*Ohler et al.* 1996 and *Gabdullin et al.* 1993]. Integrating such a model into the plume simulation would allow one to study the influence of the plume on radio signals traveling to and from the satellite.

The quasi-neutral PIC-DSMC method is not application specific and is not limited to spacecraft-plume interaction work. In principle, the model can be extended to other applications and could include additional physical effects. In the long term, the following efforts would serve to extend the regime of application for the quasi-neutral PIC-DSMC method.

- An effort needs to be undertaken to improve the electron fluid model and include the effects of the magnetic field. In the quasi-neutral formulation, the electric potential is obtained by inverting the electron momentum equation. With this particular thruster, the potential can be obtained by inverting the Boltzmann relationship. In general, however, the full electron momentum equation could be inverted using an iterative solving method. Doing this would allow the modeling of electron drift and magnetic effects. We note, however, that some previous attempts to invert the full electron momentum equation for a hybrid PIC model have been unsuccessful due to numerical instabilities [*Samanta Roy* 1995]. Developing a full electron fluid model would be a significant step forward but may require substantial time and research effort.
- An effort might be made to develop a model for Coulomb collisions. Though Coulomb interactions are not thought to have a great influence on the plume of a Hall thruster, the development of such an algorithm would be a useful tool for the modeling of plasmas in general. At the present time, the DSMC method can not be used to model short range Coulomb interactions effectively. This is because the DSMC method is not well suited to

the modeling of collisions in which only small angle deflections occur. The DSMC method is well suited to the modeling of short range, large angle deflections, so it may be possible to use an averaging method to replace many small angle deflections with an equivalent large angle deflection. Detailed work is necessary to determine if this is feasible and to determine the regimes in which such a method might apply in a plasma.

- An effort should be made to determine the role of plasma turbulence on transport properties in particle simulations. The plume of a Hall Thruster is known to exhibit considerable instability. These affect the transport coefficients in the plume, and may affect its development. This work considered only classical collisions. Further work is needed to study turbulence and its influence on the plume and to determine whether the quasi-neutral PIC-DSMC method can be modified to include turbulent effects.

In addition to the points outlined above, several general recommendations can be made with respect to the general study of spacecraft-plume interactions issues. In particular we note the following.

- Surface-interaction effects need to be studied very thoroughly. Work needs to be undertaken to measure sputtering coefficients for Xenon ions impacting on common spacecraft surfaces at normal and non-normal trajectories. Work also needs to be undertaken to identify the chemical mechanisms by which metals and ceramics can deposit on the surfaces of spacecraft.
- Better and more accurate measurements need to be taken of the very near-field plume region of Hall thrusters (< 1 cm from the thruster exit). These measurements would help improve the existing SPT-100 source model.
- Orbital tests are desirable to help validate simulations of operational conditions and to help extend laboratory tests to operational conditions.
- Experimenters should always include estimates of the accuracy of their experimental measurements when reporting plume data. Further efforts are also required to resolve inconsistencies in the existing experimental database.

Ultimately, better data of all types is needed characterize the plume and encourage the development and verification of plume-spacecraft interaction models.

Electric propulsion as a field is rapidly approaching the point of commercial viability. For years, electric propulsion has been considered a promising technology not quite mature enough for use on operational spacecraft. Arcjets have now been used on commercial communications satellites and it is very likely that both Ion and Hall thrusters will fly on operational satellites within the next three years. As EP becomes widely accepted in the satellite community, advanced models will be needed to study the interaction between plasma plumes and spacecraft. This work is a significant advance in the computational modeling of plasma plumes and will promote efforts to advance the use of EP by the spacecraft design community.

Appendix A: User's Manual for Quasi3

Version 1.0
User Manual, October 7, 1996

A.1 Packing List

The *quasi3* package consists of four programs which are contained in four separate directories. These four programs fit into two categories.

Simulation and Mesh Generation

- *quasi3* (ANSI C): the plume simulation program
- *mesh3* (ANSI C): a three dimensional embedded grid generator for *quasi3*

Visualization

- *plot3* (FORTRAN): an advanced 3-D plotting and visualization program based on the Visual3/Particle3 graphics library. It shows plume parameters in three dimensions.
- *surfplot* (C): a fairly primitive 2-D contour plotting program based on the Grafic visualization library. It shows erosion rates and other surface related data.

Each directory contains source code and a makefile. The simulation and support programs are written in ANSI C and should be fully portable to different workstation architectures.

The visualization programs are less portable. *Plot3* is written in FORTRAN and requires the visual3 library. Surfplot3 is written in C, but is linked to the Grafic library, which is written in FORTRAN. It therefore requires f77 to link properly. All of the numerical analysis is done by the simulation, so the user can also use his/her own visualization software to view the results.

The Visual3/Particle3 library is a multi-platform 3-D visualization library available from Bob Haimes at MIT. It requires GL or OpenGL. A workstation class machine is basically required to do 3D visualization, preferably a Silicon Graphics machine if it is available.

The Grafic library is a public domain visualization library. It is available on project Athena at MIT and from multiple other sources.

Before building these programs, you will probably need to edit the "FLAGS" macro in each makefile to fit your platform.

Version 1.0 is not known to contain any bugs. However, it has only been tested on a small number of platforms, so your mileage may vary. The user interface is still quite rough.

A.2 Introduction

The core of the simulation package is *Quasi3*. It takes a pregenerated grid from *Mesh3* and produces data which can be viewed and analyzed using *Plot3* and Surfplot. The sequence of files and executables is shown in Figure A.1 below.

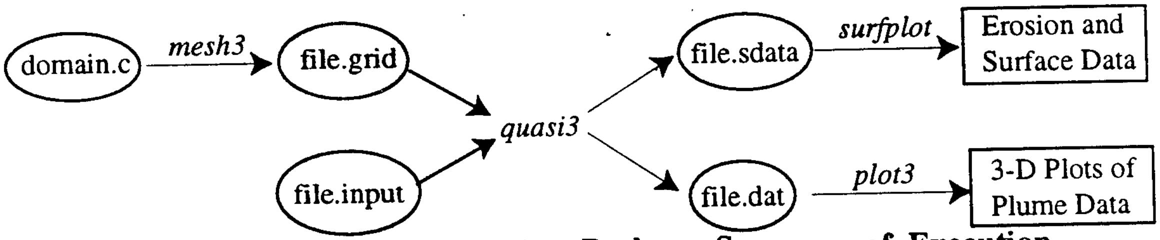


Figure A.1: Plume Simulation Package Sequence of Execution
Programs in italics, files in ovals

The spacecraft and thrusters are defined by the user in the file "Mesh3/domain.c". The program *Mesh3* is then compiled and run to produce a grid file. This grid file is the input to the simulation *Quasi3*, which produces data files for use by the *Plot3* and surfplot programs. This manual describes the usage, input and output of each program. Details of the PIC-DSMC algorithm are explained in Chapters 3 and 5. In order to construct a computational domain, the user must understand the normalization scheme used in the simulation. These are discussed in Chapter 3. The next section briefly summarizes the normalizations used by *Quasi3*.

A.3 Normalization Scheme

When conventional SI units are used for numerical analysis, large roundoff errors can occur when very large numbers (2.5 x 10^{17} m^{-3}) and very small numbers (1.6 x 10^{-19} Coulombs) are part of the same calculation. To avoid roundoff errors, *Quasi3* uses normalized (unitless) variables for its internal calculations. Both input and output quantities may use normalized variables, so a list of the normalizations and their default values is given below.

Normalized values are obtained by dividing by reference quantities, so a density of 1.0 x 10^{17} m^{-3} corresponds to a normalized density of 1 x 10^5 (see Table below). In theory, the value of various reference quantities can be controlled by the user. By selecting difference values for the four fundamental quantities, a user changes the value of all reference quantities. In practice, most users should leave these quantities alone and treat them as fixed values.

- Fundamental Quantities (set in quasi3.h and mesh3.h):
  Charge: 1.6×10^{-19} C (elementary charge)
  Density: 1.0×10^{12} m^{-3} (arbitrary)
  Mass: 2.181063×10^{-25} kg (weight of a Xe atom)
  Temperature: 2 eV (same as background electron temperature)

- Derived Quantities:
  Charge/Mass Ratio: 734,840.5 (reference mass/reference charge)
  Length: 0.010513 m (Debye length)
  Plasma Frequency: 115313.21 sec^{-1}
  Potential: 2 Volts (reference temperature in eV)
  Velocity: 1212.303955 m/s (thermal velocity)

A.4 Guide to *mesh3*

Mesh3 is a three dimensional embedded mesh generator that creates grids for use with *quasi3*. *Mesh3* sets the "layout" of objects and thrusters in the simulation domain. The input into *Mesh3* is a series of geometric parameters which are specified by editing source code in the file *Mesh3/domain.c*. By modifying these parameters, a user can lay out the entire computational domain including grid space, spacecraft geometry, and thruster location and orientation. After each modification, the code must be recompiled by typing "make" in the *Mesh3* directory.

The output from *mesh3* consists of two files with the names

file.grid: grid, object, and thruster orientation data in binary format
file.dat: grid, object, and thruster orientation data in ASCII format for use by *plot3*.

A.4.1 Command Line Interface

Mesh3 takes only one argument on the command line.

```
mesh3 -o <filename>
```

-o must be followed by a filename. It specifies the output filename for the grid files. The suffix ".grid" is automatically added to the end of the filename and should not be specified on the command line. So a command of the form

```
mesh3 -o test
```

would produce output files named "test.grid" and "test.dat".

A.4.2 Input File Format

The input to *mesh3* is actually C source code in the file *Mesh3/domain.c*. A sample file is shown on the next page and is followed by a description of each parameter.

```
/* domain.c */
/* This file is where the simulation's geometry is specified */
/* The grid resolution used to resolve an object depends on how it sits */
/* in the domain.  An object has the resolution of the finest grid */
/* in which it is ENTIRELY contained.  */

/* Objects must be at least two cells wide on some grid that contains them */
/* or the object will never be resolved enough to show on the grid */

/* Define the objects in the routine defineObjects (below) */
/* Placing an interior edge within one grid cell of an exterior boundary, */
/* or within one grid cell of another interior boundary, is definately */
/* a bad idea */

/* This configuration is part of a communications satellite */

#include <stdio.h>
#include <math.h>
#include "constants.h"
#include "mesh3.h"
#include "globals.h"

/* Widths for the simple test case grid w/ array canted at 30 degrees*/
/* Domain Definition */
#define DOMAIN_X_WIDTH 25                         /* Cells in X direction */
#define DOMAIN_Y_WIDTH 35                         /* Cells in Y direction */
#define DOMAIN_Z_WIDTH 35                         /* Cells in Z direction */

#define CELL_WIDTH 12.0                               /* Normalized units */

/* Needed to calculate thruster flow rate */
#define XE_ION_MASS (0.1313/AVOGADRO)

void defineBoundaries()
/* Sets the exterior boundary conditions (along edges of the domain) */
{
  particle_bnd_switch[NORTH] = ABSORBING;
  particle_bnd_switch[SOUTH] = REFLECTIVE;
  particle_bnd_switch[EAST] = ABSORBING;
  particle_bnd_switch[WEST] = ABSORBING;
  particle_bnd_switch[UP] = ABSORBING;
  particle_bnd_switch[DOWN] = ABSORBING;

  neutral_flux_switch[NORTH] = FALSE;
  neutral_flux_switch[SOUTH] = FALSE;
  neutral_flux_switch[EAST] = FALSE;
  neutral_flux_switch[WEST] = FALSE;
  neutral_flux_switch[UP] = FALSE;
  neutral_flux_switch[DOWN] = FALSE;

  ion_flux_switch[NORTH] = FALSE;
  ion_flux_switch[SOUTH] = FALSE;
  ion_flux_switch[EAST] = FALSE;
  ion_flux_switch[WEST] = FALSE;
  ion_flux_switch[UP] = FALSE;
  ion_flux_switch[DOWN] = FALSE;
}

void defineEmbeddedGrids()
{
  nembeddedgrid = 1;         /* nembeddedgrid is a global variable */
  embeddedgrid = (embeddeddata *) calloc (nembeddedgrid, sizeof(embeddeddata));

  /* Piece 1: a box */
  if (nembeddedgrid > 0) {
    embeddedgrid[0].xc = 6;       /* Width in grid points on parent mesh */
    embeddedgrid[0].yc = 6;
    embeddedgrid[0].zc = 5;
    embeddedgrid[0].xoffc = 9;    /* Offset in grid points on parent mesh */
    embeddedgrid[0].yoffc = 6;
    embeddedgrid[0].zoffc = 14;
    embeddedgrid[0].parent = -1;  /* -1 indicates the top mesh is the parent */
  }
}

void defineObjects()
{
  nobjects = 3;         /* nobjects is a global variable, type short int */
  objects = (box *) calloc (nobjects, sizeof(box));

  /* Piece 1: spacecraft main body */
  objects[0].index = 0;                     /* Grid in which coord are given */
                            /* This grid must surround the object ENTIRELY */
  objects[0].xc = 10;                                     /* X-width, coord. */
  objects[0].yc = 8;                                      /* Y-width, coord. */
  objects[0].zc = 21;
  objects[0].xoffc = 0;
  objects[0].yoffc = 0;
  objects[0].zoffc = 2;
  objects[0].type = FLOATING;                /* Object surface boundary type */
  sprintf(objects[0].name, "bus");              /* NO SPACES ALLOWED in name */

  if (nobjects > 1) {
    /* Piece 2: solar array */
    objects[1].index = 0;                 /* Grid in which coord are given */
                          /* This grid must surround the object ENTIRELY */
    objects[1].xc = 9;                                    /* X-width, coord. */
    objects[1].yc = 12;                                   /* Y-width, coord. */
    objects[1].zc = 1;
    objects[1].xoffc = 0;
    objects[1].yoffc = 22;
    objects[1].zoffc = 12;
    /* Object surface boundary type */
    objects[1].type = FIXED;          /* Object surface boundary type */

    sprintf(objects[1].fixedfilename[NORTH],
           "Configs/Potential/intelsat1.halfarray.n");
    sprintf(objects[1].fixedfilename[SOUTH],
           "Configs/Potential/intelsat1.halfarray.s");
    sprintf(objects[1].fixedfilename[EAST],
           "Configs/Potential/intelsat1.halfarray.e");
    sprintf(objects[1].fixedfilename[WEST],
           "Configs/Potential/intelsat1.halfarray.w");
    sprintf(objects[1].fixedfilename[UP],
           "Configs/Potential/intelsat1.halfarray.u");
    sprintf(objects[1].fixedfilename[DOWN],
           "Configs/Potential/intelsat1.halfarray.d");
    sprintf(objects[1].name, "solar_panel");/* NO SPACES ALLOWED in name */

  }

  if (nobjects > 2) {
    /* Piece 2: yoke for array */
    objects[2].index = 0;                 /* Grid in which coord are given */
                          /* This grid must surround the object ENTIRELY */
    objects[2].xc = 1;                                    /* X-width, coord. */
    objects[2].yc = 15;                                   /* Y-width, coord. */
    objects[2].zc = 1;
    objects[2].xoffc = 0;
    objects[2].yoffc = 8;
    objects[2].zoffc = 12;
    objects[2].type = FLOATING;            /* Object surface boundary type */
    sprintf(objects[2].name, "yoke");         /* NO SPACES ALLOWED in name */
  }
}

void defineSources()
/* Modified for 3-D */
/* The source location is given by its center.  So, if you place the source */
/* on a flat edge, you can end up getting                                   */
/* particles showing up INSIDE the object it's attached to (which is        */
/* obviously a bad thing).  So, be cautious in how you use these sources    */
/* at least for the near future */
/* 8/7/96, -David Oh (bamf@mit.edu) */
{
  nsources = 1;
  sources = (sourcedata *) calloc (nsources, sizeof(sourcedata));

  if (nsources > 0) {
    /* Piece 1: a hall thruster half */
    sources[0].nobj = 0;                            /* Object it's attached to */
    sources[0].xc = 115.911;      /* Coordinates on object (normalized units */
    sources[0].yc = 100.756;
    sources[0].zc = 157.058;

    /* Vector pointing along the thruster exit direction */
    sources[0].thrustx = 0.25;
    sources[0].thrusty = 0.9659;
    sources[0].thrustz = 0.0670;

    /* Vector perpendicular to thrust vector pointing from center to cathode */
    sources[0].cathodex = -0.9330;
    sources[0].cathodey = 0.2588;
    sources[0].cathodez = -0.25;

    /* Total propellant flow rate in #/sec (cathode and anode) */
    sources[0].flow_rate = (5.37e-6/XE_ION_MASS);
  }
}

void initializeVariables()
/* Sets globals to values specified in define statements at the top of */
/* domain.c */
{
  domain_x_width = DOMAIN_X_WIDTH;
  domain_y_width = DOMAIN_Y_WIDTH;
  domain_z_width = DOMAIN_Z_WIDTH;
  cell_width = CELL_WIDTH;
}
```

A.4.2.1 *Mesh3* Parameters

DOMAIN_X_WIDTH
DOMAIN_Y_WIDTH
DOMAIN_Z_WIDTH
CELL_WIDTH

The first three parameters specify the dimensions of the domain in cells. The CELL_WIDTH parameter gives the width of each cell in normalized units. In the example case, the cell width is 12 and the domain x, y, and z widths are 25, 35 and 35 respectively. The dimensions of the domain are therefore given by:

X: DOMAIN_X_WIDTH x CELL_WIDTH x ref_length =
25.0 x12.0 x 0.010513 m =3.15 m
Y: DOMAIN_Y_WIDTH x CELL_WIDTH x ref_length =
35.0 x12.0 x 0.010513 m =4.42 m
Z: DOMAIN_Z_WIDTH x CELL_WIDTH x ref_length =
35.0 x12.0 x 0.010513 m =4.42 m

XE_ION_MASS

This is mass of a Xe ion in kg. It's used for convenience and shouldn't need to be modified. It is used only in domain.c and is not referenced in by *mesh3*.

The *defineBoundaries()* routine is where users specify boundary conditions. Three boundary conditions are specified for each of the six faces of the domain. The faces are referred to by direction, as shown in Figure 2. So North is the positive y face, south is the negative y face, east is the positive x face, etc.

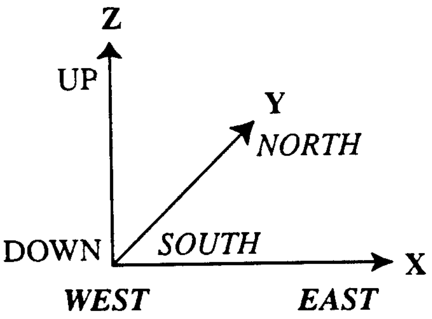


Figure A.2: Directions and Axises

The three boundary conditions are

particle_bnd_switch[direction] = [ABSORBING || REFLECTIVE];

This switch specifies what happens when a macro particle strikes the face. When set to ABSORBING, *quasi3* deletes particles that hit the face. When set to REFLECTIVE, *quasi3* reflects particles back into the domain. REFLECTIVE boundary conditions should be used only when a boundary is a plane of symmetry. Since the file is written in C; you must use all caps to specify the boundary condition.

neutral_flux_switch[direction] = [TRUE || FALSE];
ion_flux_switch[direction] = [TRUE || FALSE];

When true, an inward flux of neutrals/ions is present along the face. When false, no flux is present. The magnitude and velocity of the flux is determined from the background neutral/ion temperature and density. The neutral flux switch should be set to TRUE when the simulation includes an ambient neutral background (in, for instance, a laboratory test), FALSE when simulating a spacecraft in orbit. The ion flux switch is basically inoperative and should be set to FALSE all the time.

The defineEmbeddedGrids() routine is where users specify the position and dimensions of embedded grids. The use of embedded grids is a tricky thing in particle based simulation. In a conventional CFD code, a user would increase the grid resolution (use smaller cells) in order to increase the accuracy of the simulation. In a PIC-DSMC code, increasing the grid resolution without increasing the number of particles present actually decreases local accuracy by decreasing the number of particles in each cell. However, smaller cells are necessary to resolve areas with high gradients or other important details. We recommend using one level of grid to resolve a small area just outside the thruster exit and avoiding its use elsewhere in the domain. The macro particle weights should be adjusted so that there are 10 particles per cell within the embedded mesh as well as throughout the rest of the domain.

The location and size of each embedded grid is defined in terms of its parent grid. A parent is a grid which is one size larger than the embedded grid and fully contains the embedded grid There are several limitations on where an embedded mesh can be placed on a parent mesh.

- An embedded mesh can not intersect a mesh which is two levels larger than itself, so the parent mesh must include a margin of one cell around each face of the embedded grid unless the edge is backed up against the outside of the domain (see Figure 3 below).
- An embedded mesh must be assigned to one and only one parent, so an embedded grid can not overlap two parent grids.
- An embedded mesh can not intersect another embedded mesh with the same resolution.
- An embedded grid has a unique identification number assigned to it by the user. The grids must be numbered continuously from 0 to (the total number of grids-1). These values are used to assign parents to embedded meshes. Therefore, an embedded mesh can not have an identification number which is less than its parent's identification number.

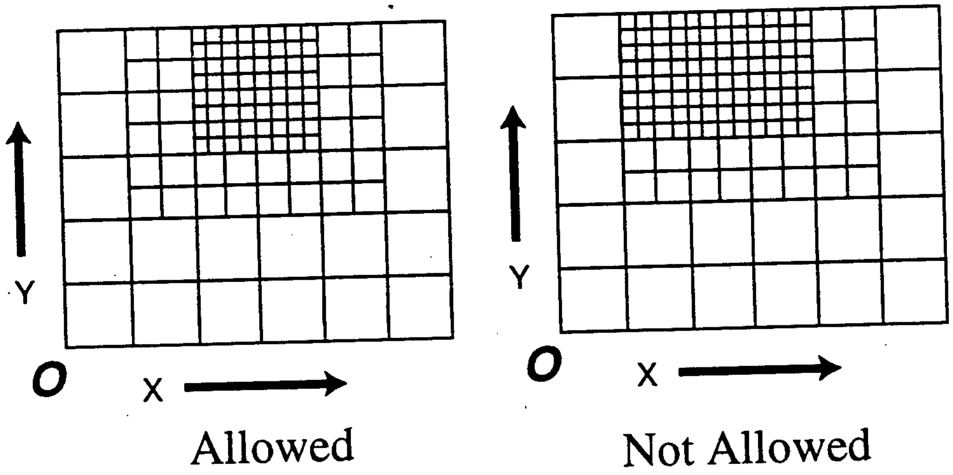


Figure A.3: Permitted and Forbidden Placement of an Embedded Mesh

A parent *can* have more than one embedded mesh. The various embedded mesh control parameters are

nembeddedgrid = [number]

This parameter specifies the number of embedded meshes present in the simulation. Each embedded grid is refered to by a unique number. The identification numbers are assigned sequentially starting from zero, so the first grid would be grid 0, the second grid 1, etc. The top level mesh which defines the domain is automatically assigned the value -1. We will refer to the local mesh's identification number as *gridid.*

Every child mesh must be assigned the following two parameters in domain.c. They give the grid's size and location.

embeddedgrid[*gridid*].xc = [length in cells (integer)];
embeddedgrid[*gridid*].yc = [length in cells (integer)];
embeddedgrid[*gridid*].zc = [length in cells (integer)];

The three parameters above give the dimensions of the embedded mesh in cells of the parent mesh (see Figure 3 below). The lengths must be integer values. Non-integer values will be rounded down. xc, yc, and zc give the dimensions in the x, y, and z directions respectively.

embeddedgrid[*gridid*].xoffc = [offset in cells (integer)];
embeddedgrid[*gridid*].yoffc = [offset in cells (integer)];
embeddedgrid[*gridid*].zoffc = [offset in cells (integer)];

The three parameters above give the offset of the origin of the embedded mesh from the origin of the parent mesh in cells. The origin is defined to be the lower-south-west corner of the mesh (as shown in Figure 2). A two dimensional example is given in Figure A.4.

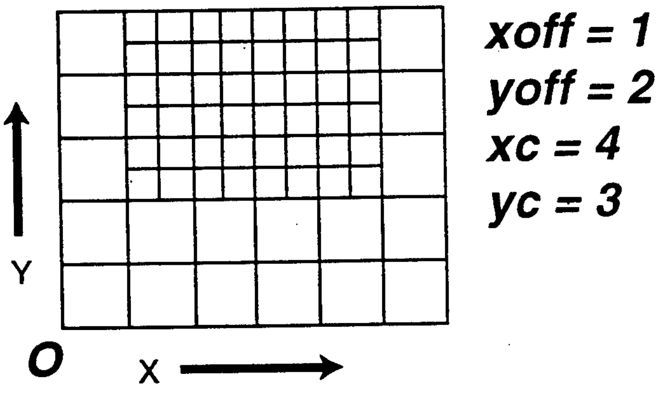


Figure A.4: Embedded Mesh Placement Coordinates

embeddedgrid[*gridid*].parent = [*nparent*];

This parameter designates the local grid's parent mesh by its identification number. *nparent* must be an integer less than *gridid* . A value of -1 indicates that the parent mesh is the top mesh in the simulation (the one which defines the dimensions of the domain itself).

The *defineObjects()* routine is where users designate the size and position of objects in the domain. As with the embedded meshes, there are certain restrictions on where objects can be placed in the domain.

- Every face of every object must be at least one cell away from parallel faces of embedded meshes, as shown in Figure A.5 below. It is fine to have embedded meshes overlapping objects as long as the face of the mesh doesn't lie exactly on the face of the object.
- Objects should not overlap each other, or unpredictable results may occur.
- Objects can be placed directly adjacent to each other. However, if there is no gap between the objects, the current density, neutral density, and potential will be miscalculated along the junction between the two objects. This is a problem which results from an incorrect value for the weighting factor. Generally, this problem can be safely ignored, but values calculated at these junctions should not be taken seriously.

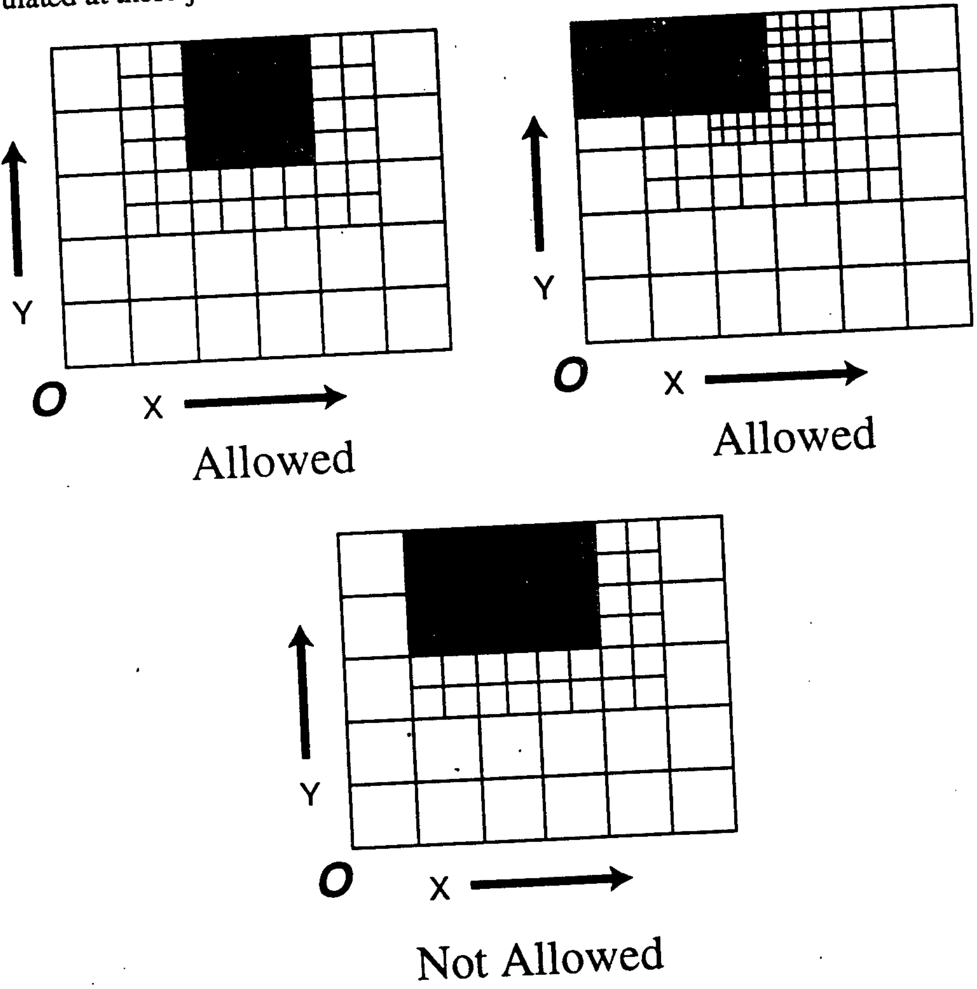


Figure A.5: Permitted Object Placement

Like the embedded meshes, each object is assigned a unique identification number which is given sequentially starting with zero. We will refer to this value as *objectid. The* following parameters are defined in defineObjects().

nobjects = [*integer*];

This parameter gives the number of objects present in the domain. In the example case, it is set to three because there are three objects present: the satellite body, solar array, and yoke.

objects[*objectid*].index = [*integer*];

This parameter is the *gridid* of the parent grid which completely contains the object. The parent grid must entirely surround the object with a margin of at least one cell between the edges/faces of the object and the edges/faces of the parent grid. A value of -1 indicates that the parent grid is the top level grid.

objects[*objectid*].xc = [length of object in cells (integer)];
objects[*objectid*].yc = [length of object in cells (integer)];
objects[*objectid*].zc = [length of object in cells (integer)];

These give the dimensions of the object in cells on the parent mesh, using the same method used to give the dimensions of embedded meshes (see Figure 4 above). The values must be integer values. Non-integer values will be rounded down. xc, yc, and zc give the dimensions in the x, y, and z directions respectively.

objects[*objectid*].xoffc = [offset in cells (integer)];
objects[*objectid*].yoffc = [offset in cells (integer)];
objects[*objectid*].zoffc = [offset in cells (integer)];

These parameters give the offset of the origin of the object from the origin of the parent mesh in cells. The origin is defined to be the lower-south-west corner of the mesh (as shown in Figure A.2). xoffc, yoffc, and zoffc give the offset in the x, y, and z directions respectively. A two dimensional example is given in Figure A.4. The example shown is an embedded mesh, not an object, but the same system is used in both cases.

objects[*objectid*].type = [FLOATING || FIXED];

This specifies the potential boundary condition to be applied on the surface. If the object is designated as "FLOATING", the surface potential is calculated by balancing the ion and electron flux to the surface of a non-conductive object. If the object is designated as "FIXED", the surface potential is set at a fixed value. The fixed potential profile is determined from input files specified in the object[*objectid*].fixedfilename parameter.

Optional parameter: objects[*objectid*].fixedfilename[direction];

This designates the file from which the fixed potential profile is to be read. It can be ignored with FLOATING objects but is required with FIXED ones. The maximum length of the file name is 99 characters.

sprintf(objects[*objectid*].name, *string*);

This gives the object a name that the user can use to identify it when using surfplot. The name must be a string with no more than 50 characters and containing no spaces. The name is not used by *quasi3*. It is just there to help identify interesting objects.

The *defineSources()* routine is where users specify the location and orientation of thrusters in the domain. As with objects and embedded meshes, every source has a unique *sourceid*. A list of parameters follows.

nsources = [integer];

This gives the number of thrusters present in the domain. It should work for any number of thrusters, but has never been tested with more than one.

sources[*sourceid*].nobj = [integer];

This parameters (nobj) gives the *objectid* of the object the source is attached to (its parent). Every source must be associated with an object, though its physical location may not necessarily be directly on the surface of the parent object.

sources[*sourceid*].xc = [float];
sources[*sourceid*].yc = [float];
sources[*sourceid*].zc = [float];

These parameters give the location of the center of the anode's exit plane as measured from the origin of the object(*objectid*) in <u>normalized</u> units. The values can be anything, but should be carefully set so the thruster's exit plane does not end up partially or entirely inside the parent object. Though the simulation will operate in this mode, the results are (obviously) useless.

sources[*sourceid*].thrustx = [float];
sources[*sourceid*].thrusty = [float];
sources[*sourceid*].thrustz = [float];

These parameters define the x, y, and z components of a vector which is parallel to the thrust vector points in the flow exit direction (as shown in Figure A.6). *Quasi3* uses the thrust vector to determine the direction of flow. The vector can have an arbitrary magnitude.

sources[*sourceid*].cathodex = [float];
sources[*sourceid*].cathodey = [float];
sources[*sourceid*].cathodez = [float];

These parameters define the x, y, and z components of a vector which is perpendicular to the thrust vector and points from the center of the anode exit towards location the cathode (as shown in Figure A.6). *Quasi3* uses the cathode vector to determine where the cathode should be placed relative to the anode. The vector can be of arbitrary magnitude but should be perpendicular to the thrust vector, or unpredictable things may occur.

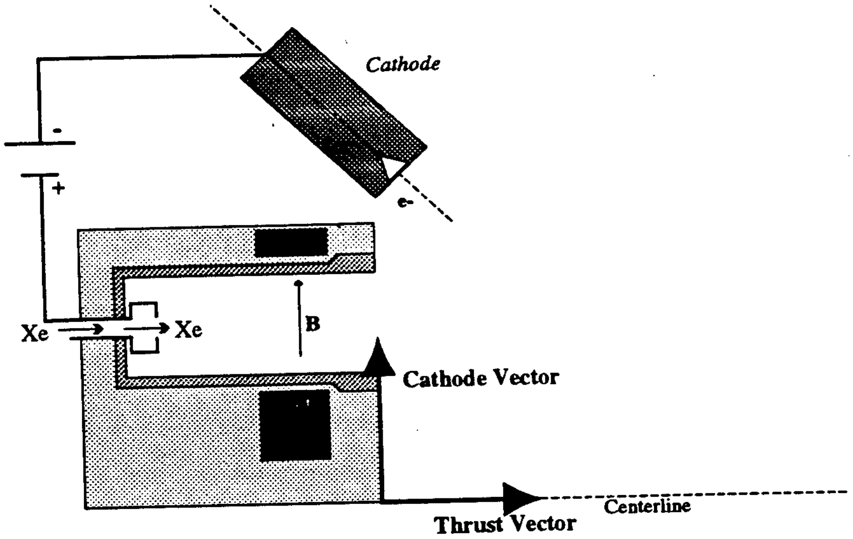


Figure A.6: SPT Orientation Vector

sources[*sourceid*].flow_rate = [Xe mass flow rate in molecules/second (float)];

Gives the total propellant flow rate (both cathode and anode) through the thruster in number of molecules per second. Additional flow may be added by changing the *anode_prop_injection_frac* parameter in *file.input* (see *quasi3* below).

The routine *initializeVariables()* is used by *quasi3* and should not be modified by the user.

A.4.3 Output Files

mesh3 creates two output files.

> *file.grid*: grid, object, and thruster orientation data for *quasi3* (binary format)
> *file.dat*: three dimensional visualization data for *plot3* (ASCII format)

Note that *file.grid* is in binary format, and may not be transferrable across platforms (from SGI to Dec workstations, for instance). *file.dat* is available so the user can quickly examine the domain using *plot3*. It is not used by *quasi3*.

A.4.4 Notes

When objects are placed adjacent to each other, the current density, neutral density, and potential will be miscalculated along the junction between the two objects. This is a problem which results from an incorrect value for the weighting factor. Generally, this problem can be safely ignored, but values shown at the junction should not be taken seriously.

A.5 Guide to *quasi3*

As indicated in Figure A.1, *quasi3* is the heart of the plume simulation package. Like *mesh3*, *quasi3* uses a command line interface to specify filenames and determine the number of iterations to execute in each run. The computational domain is specified in two input files. These files are:

file.grid: grid, object, and thruster orientation data from *mesh3* (binary format)
file.in: input parameters specific to each run (ASCII format)

("file" is a file prefix which is set on the command line)

When done, *quasi3* creates a series of output files which contain the simulated data. These files are:

file.bin: data necessary to restart the simulation from its last iteration (binary format)
file.err: error and progress log (ASCII format)
file.num.dat: three dimensional plume data in *plot3* format (ASCII)
file.num.sdata: two dimensional surface data in surfplot format (ASCII)
file.num.cur: Debugging data; given the ion current density vs. angle at a fixed distance from the thruster exit. This radius is defined in the file "domain.h." and can change. (ASCII)
file.num.arcsurface: Debugging data; this is binary data that can be used by the 2-D surface post-processor to estimate the sputtering which would occur along a virtual surface locationed at a fixed distance from the thruster exit. This radius is defined in the file "domain.h" and can change. (Binary)

("file" indicates a file prefix which is set on the command line)
("num" is the iteration number on which the file was created)

The visualization data files are in ASCII format, so they can be taken from machine to machine. The actual restart file (.bin) is in binary format, so you generally can't start a run on one platform and restart it on another one.

A.5.1 Command Line Interface

The file prefix is one of many options specified on the command line. A summary of command line options is below.

```
quasi3 [-new -file <filename> -gridfile <filename> -iterations <num> -outputfileprefix
<filename> -restartfile <num> -saveinterval <num>]
```

-new is a switch:

Tells *quasi3* to start a new run rather than use a restart file. When -new is not present, *quasi3* looks for a restart file with the name [outname].bin (see the -outputfile option below).

-file must be followed by a filename

The name of the input data file used when starting a new run. It is required with *-new* but ignored in restart mode (the default mode). The name of the input file must end with ".in". This suffix is automatically added and should not included on the command line. An example:

```
quasi3 -n -file Input/test
```

would go to the directory named "Input" and load the file "test.in".

If *-file* is not present and *-new* is specified the program will use the default filename Input/q3.in.

-grid must be followed by a filename

-grid is the name of the grid file used for the domain when starting a new run. It is required with *-new* and ignored in restart mode (the default mode). The name of the grid file must end in ".grid". This suffix is <u>automatically</u> added and should not be included on the command line. An example:

```
quasi3 -n -grid test
```

would load the file "test.grid".

If *-grid* is not present and *-new* is specified, the program will use the default name "test.grid".

-iter must be followed by a number

-iter is the number of iterations to execute. If *-iter* is not-present, the program will do 100 iterations before finishing.

-outfileprefix must be followed by a string

-outfileprefix specifies the file prefix for the restart and output files. If *-outfileprefix* is not there, the program will choose the default file prefix "Data/v3out".

-restart must be followed by a number

-restart specifies the number of iterations between saves of restart data. If *-restart* is not present, the *quasi3* will save a restart file every 500 iterations. A restart file is also saved at the end of a run, regardless of how many iterations have taken place. This option provides the user with ongoing backups in case the program/computer crashes during the course of a run.

-save must be followed by a number

-save is the number of iterations between saves of visualization data files. If *-save* is not present. *quasi3* will dump data every 5000 iterations.

quasi3 distinguishes options by their first letter, so only this first letter are necessary. In other words:

```
quasi3 -n -f Input/q3 -i 500 -g Grid/q3
```

is equivalent to

```
quasi3 -new -file Input/q3 -iteration 500 -gridfile Grid/q3
```

A.5.2 Input File Format

A user formatted input file is used to specify parameters which are independent of the grid and can vary from run to run. The file consists of a series of lines of the form `parameter_name: value` and the filename must end in ".in". A sample input file is shown below.

```
/* This is a sample file for the user manual */
/* May 29, 1996 */
bkg_neutral_density: 7.085e16          /* Molecules/Meter Cubed */
anode_prop_ingestion_fraction: 0.124
anode_double_ion_fraction: 0.2
xe_ion_wt: 1.25e11                 /* Real particles/macro particle */
xe_2_ion_wt: 3.0e10                /* Real particles/macro particle */
xe_neutral_wt: 8.0e12              /* Real particles/macro particle */
```

Quasi3 sets each parameter equal to the value specified after the colon. The order of the parameters is irrelevant as long as each ends with a colon followed by a space. *Quasi3* recognizes C-style comments and ignores everything between "/*" and "*/".

A.5.2.1 *Quasi3* Parameters

anode_double_ion_fraction (Dimensionless) specifies the fraction of the ions leaving the thruster which are double ions. The rest of the ions are single ions. For an SPT-100, the best data available indicates that this parameter should be set to 0.2 [*Manzella* 1993].

anode_prop_ingestion_fraction (Dimensionless) is used only when simulating tank experiments. In some experiments, the total measured current actually exceeds the current put out by the thruster (as discussed in Chapter 4). Some of this current is due to the ingestion of ambient neutrals into the thruster and some is due to other sources. This switch is a means of compensating for measured differences between the thruster's nominal flow and measured increases in the total current. A value of zero means that no ambient neutrals are being ingested. A value of one indicates that the effective propellant flow is twice the nominal flow. When simulating an experiment, this parameter should be set equal to (total measured current-nominal current)/(nominal current). When simulating a spacecraft, this parameter should be set equal to zero.

bkg_neutral_density (Molecules/Meter Cubed) gives the background neutral density in the simulation. When simulating an experiment, this value would be determined from the tank pressure. When simulating a spacecraft in GEO, this is value would typically be set equal to zero.

xe_ion_wt (Dimensionless) is the number of real molecules represented by each Xenon ion macro particle. The key here is to choose a value which results in, on average, 10 macro particles per cell in regions of interest in the domain. This includes the core region of the SPT plume, but does not include the edges of the plume. The number of particles needed is discussed in more detail in Chapter 6.

xe_2_ion_wt (Dimensionless) is the number of real molecules represent by each Xenon double ion macro particle. As with *xe_ion_wt*, the key is to have at least 10 macro particles per cell. In general, if xe_ion_wt is set correctly and *anode_double_ion_fraction* is equal to 0.2, *xe_2_ion_wt* should also be 0.2 times *xe_ion_wt*.

xe_neutral_wt (Dimensionless) is the number of real molecules represented by each neutral Xe macro particle. As with *xe_ion_wt*, the key is to have at least 10 macro particles per cell.

A.5.3 Notes

Unlike conventional finite-difference and finite-element methods, particle simulation methods are unconditionally numerically stable. If abused, *Quasi3* can return absolute nonsense with no obvious indication to the user. Take a good look at what you get and see it if makes sense.

A.6 Guide to *plot3*

A.6.1 Command Line Interface

Plot3 takes only one argument on the command line.

```
plot3 <filename>
```

where `<filename>` specifies the name of the input file. The suffix ".dat" is automatically added to the end of the filename and should not be specified on the command line. So a command of the form

```
plot3 test
```

would read the file named "test.dat".

A.6.2 Input Files

Plot3 reads the visualization file produced by *quasi3* with the title "file.dat". This is the only input file required by *plot3*.

A.6.3 Instructions

plot3 uses the standard visual3/particle3 interface. The controls are described in the *visual3* and *particle3* manuals. Hitting "?" in any of the four windows results in a brief list of keys and what they do. Some keymappings which are specific to *plot3* are listed below.

d: shows the log (base 10) of the normalized charge density
D: shows the log (base 10) of the time averaged charge density
i: shows the number of single ions macro particles/cell
I: shows the number of double ions macro particles/cell
n: shows the log (base 10) of the neutral density in normalized units
N: shows the log (base 10) of the time average neutral density in normalized units
p: shows the normalized potential
s: color particles by their species
u: color particles by x-velocity
v: color particles by y-velocity
w: color particles by z-velocity
[: shows the normalized time averaged potential

The following key indicates the species of different particles

0: CEX Xenon Single Ion
1: Source (beam) Xenon Ion
2: Background Xenon Ion (generally not used)
3: Source (beam) Xenon Double Ion
4: CEX Xenon Double Ion
5: Source Xenon Neutral
6: Background Xenon Neutral

The interface takes some getting used to.

A.6.4 Notes

A large amount of RAM may be required when visualizing very large simulations (> 100,000 particles). The amount of RAM used by the program changes with time. Keep track of how much the program is using.

Turning what you see on the screen into journal quality pictures can be difficult. The easiest thing to do is take a snapshot of the windows you would like to duplicate using a program like xv. The snapshot can be saved in gif format and transferred over to the Macintoshes using fetch (an ftp program). As long as the filename ends in ".gif", fetch will correctly transfer the file in binary mode and set it to an appropriate file type. The ".gif" file can be read using Photoshop, edited as necessary (redoing scales, adding references, etc.), and then saved as a PICT file. The PICT file can be directly included in Word and Powerpoint documents.

A.7 Guide to *surfplot*

surfplot is a very basic plotting utility. The file formats are simple (see input files below), so we recommend that you build yourself your own viewer using your favorite software, and then send us a copy so we can use it too.

A.7.1 Command Line Interface

surfplot only requires one argument on the command line.

```
        surfplot {-g -b} <filename>
```

where `<filename>` gives the name of the data file. The suffix ".sdata" is automatically added to the end of the filename and should not be specified on the command line. So a command of the form

```
        surfplot test
```

would read the file named "test.sdata". The -g and -b flags specify whether the contours should be printed in color, greyscale, or black and white. The default is color. -g gives a greyscale plot. -b gives a black and white plot.

A.7.2 Input File Format

surfplot reads the surface data file produced by *quasi3*. The file has the title "file.sdata" and is the only input file required by surfplot. The surface data file uses an ASCII format and contains the following information in the following format. "\n" represents a return, "\t" represents a tab.

nobjects \n

The first entry is an integer giving number of objects present. It is followed by a return.

nsputteredmaterials\n

Number of materials on which sputtering calculations were performed. (Typically 3: Silver, Silicon, and Quartz).

material_name \n

A string giving the name of the sputtered material, followed by a return. The name will contain no spaces. There is a name given for each of the *nsputteredmaterials*.

For each of the objects present in the domain, a bunch of different data is written out. The first element is the name of the object, followed by a return.

object_name\n

Each side of the object has a separate set of data associated with it. The file starts with the NORTH side of the domain.

imax \t *jmax* \n

Gives the dimension, in nodes, in the i direction and the j direction. i always corresponds to the coordinate that comes first alphabetically. So on the UP and DOWN faces, i is the x direction and y is the y direction. On the NORTH and SOUTH faces, i is the x axis and j is the z axis. On the EAST and WEST faces, j is the y axis and k is the z axis.

The nodes are then written out in rows (i.e. in the i direction first, then the j direction, as would be given by

```
for (j = 0; j < jmax; ++j)
  for (i = 0; i < imax; ++i) { i, j, ... }
```

For each node, the following values written to disk.

i \t *j* \t

The local coordinates of the node (i.e. node 3,4).

mean_j \t

Mean current density to the surface at that node.

flux[XE_ION] \t *mean_energy*[XE_ION] \t *std_energy*[XE_ION] \t *particle_count*[XE_ION] \t

The flux, mean energy (eV), and standard distribution in the impact energy of the Xe ions on the surface at this point, followed by the number of macro particles which actually struck the surface.

flux[XE_2_ION] \t *mean_energy*[XE_2_ION] \t *std_energy*[XE_2_ION] \t *particle_count*[XE_2_ION] \t

flux[XE_NEUTRAL] \t *mean_energy*[XE_NEUTRAL] \t *std_energy*[XE_NEUTRAL] \t *particle_count*[XE_NEUTRAL] \t

The same quantities for Xe double ions and Xe neutral atoms.

potential \t *sheath_potential* \t

The potential of the surface (Volts) and the potential drop/rise across the sheath (also in Volts).

sputter_depth[0] \t *sputter_depth*[1] \t ... *sputter_depth*[nsputteredmaterials-1] \n

Finally, for each of the sputtered materials, an erosion depth is written out in microns/200 hours.

After all of the nodes on the NORTH side have been written to disk, *quasi3* proceeds to the other sides in the following order: NORTH, EAST, SOUTH, WEST, UP, DOWN.

Finally, after completing the first object, *quasi3* writes the other objects using the same format.

A.7.3 Instructions

Surfplot uses a menu interface which is intended to be self explanatory. The first menu asks the user to select an object based on the name given to it in *domain.c* (see *Mesh3*). The next menu asks the user to select a face based on its direction (as shown in Figure 2). The final menu asks which parameter to display and then creates the contour plot. A contour plotting window appears immediately afterwards.

The horizontal axis is always the coordinate which comes first alphabetically with its origin on the left hand side. The vertical axis is the other coordinate with its origin on the bottom. The axises are labeled, and one should carefully pay attention to the orientation of the plot

The contour window uses standard controls described in the Grafic manual. Professor Daniel Hastings is familiar with this graphics package and should be able to help you with the controls. Some useful keybindings are

b: show contour value at pointer
c: draw contour
g: show grid
r: redraw screen (useful when those irritating black patches appear)
x: close window

When done viewing, hit x to close the window.

A.7.4 Notes

Sputtering rates are given in units of microns/month (720 hours).

Converting surfplot images to journal quality images is a challenge. Converting them to color images is relatively easy. Use the program xv to take an image of the surfplot window and follow the procedure discussed at the section A.6.4. To produce greyscale images, use the following procedure.

>Execute surfplot with the -g option, i.e. "*surfplot -g intelsat.45.45.15000*" and bring up the plot you want to make into a figure. You should end up with a greyscale contour plot.
> Execute xv, i.e. "*xv &*"
> In the xv window, click the right mouse button. An xv control window should appear.
> In the control window, click on the grab button with the left button. This will bring up the grab window.
> In the grab window, read the directions, then çlick the grab button with the left mouse button. The grab window will disappear.
> Click in the xv contour window. The main xv window should flash, and then an image of the contour map should appear.
> In the control window, click on the save button (left mouse)
> Set the FORMAT: to GIF and COLORS: to greyscale.
> Save the file in the appropriate directory. IMPORTANT: make sure the file name ends in .gif!
> Log out and Goto a macintosh. (don't forget to write down your filename!)
>Open the program "fetch"
> From the file menu, open a new connection to the machine on which the .gif file is stored.
> Download the file from the workstation to the macintosh.
> Quit fetch.
> Open Photoshop
> From the file menu, open the file you just uploaded.
> After the file appears, goto the "Mode" menu and execute the "Color Table..." command.
> When the Color Table window appears, click on the load button.
> Load the "Grafic Contour Plot Color Table". (if you don't have a copy on the macintosh, you can download it from the documentation directory; there should be a copy here in binhex format).
> Goto the file menu and click on "save as..."
> For FORMAT: select "PICT File".
> Save the file. now you can import it directly into Microsoft Word 5.1, Microsoft Word 6.0, or Powerpoint.

Appendix B: Additional Three Dimensional Results

Appendix B shows additional results from three dimensional simulations of the SPT-100 mounted on a geosynchronous communications satellite. The results presented are contour plots of planar cuts through the three dimensional domain. Further details about the thruster and computational domain simulated are given in Chapter 6.

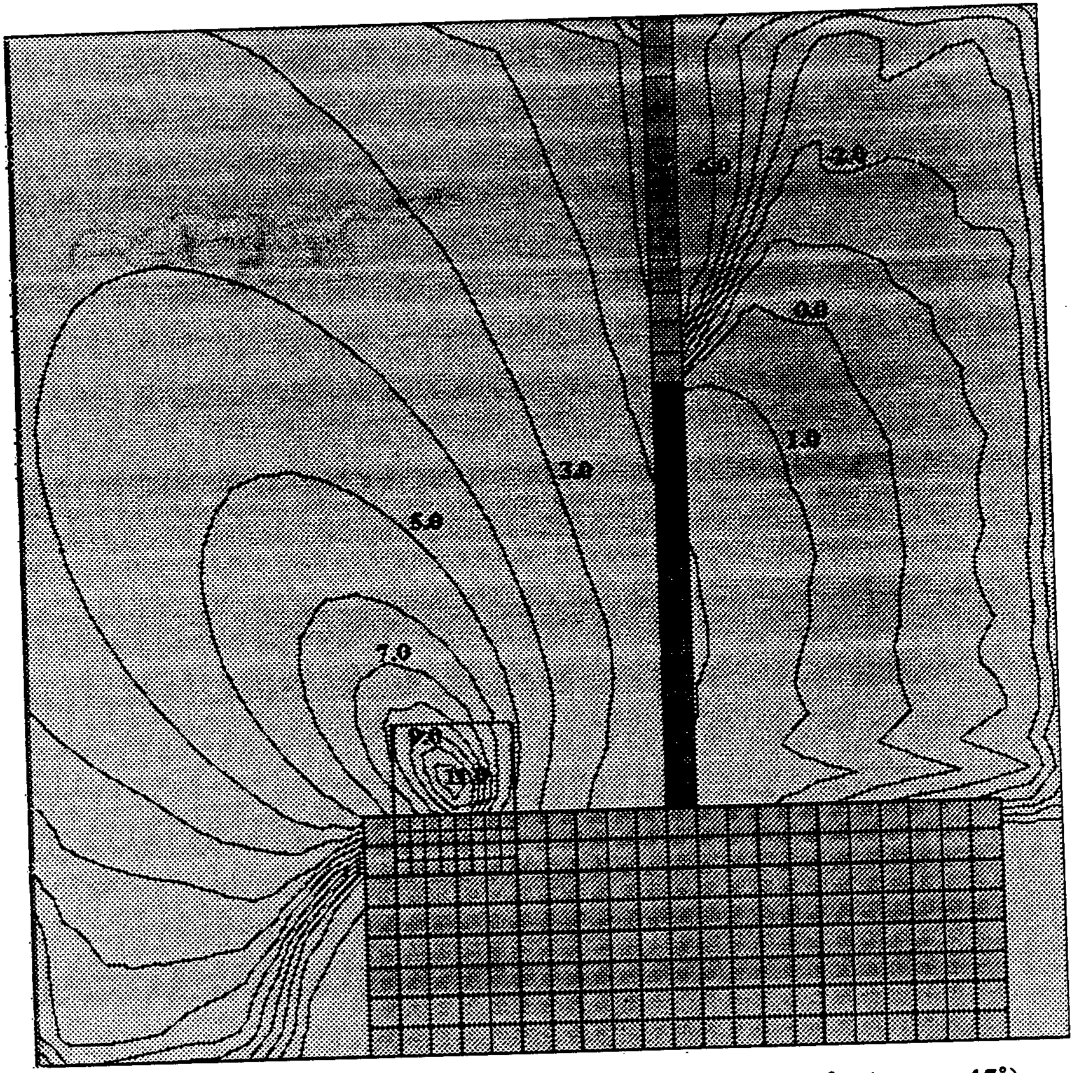


Figure B.1: Contour Plot of Potential (Cant 45°, Array 45°)
y/z Planar Cut Through Center of Thruster
Normalized Units, 20 Contours from -7 to +12

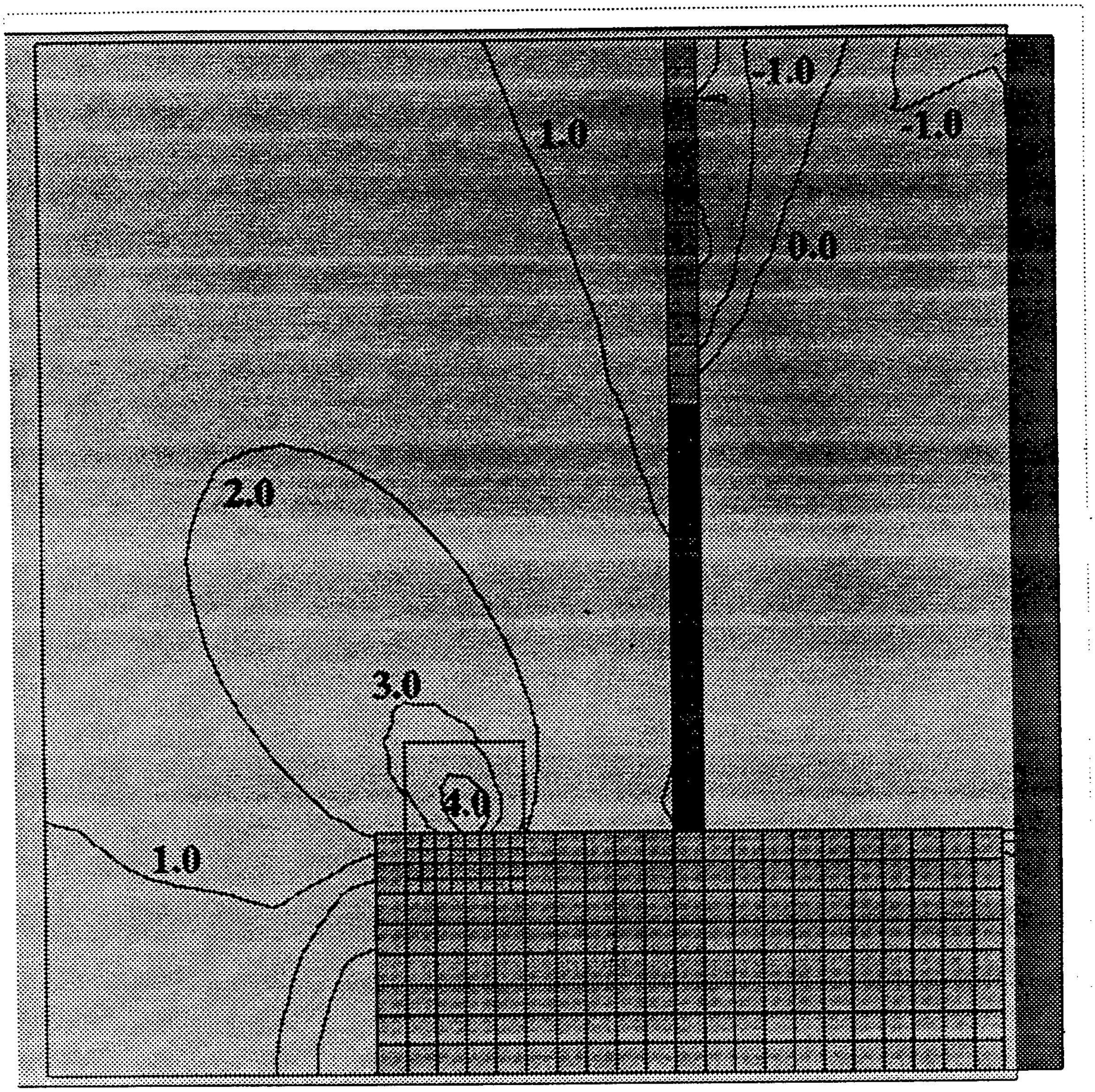


Figure B.2: Contour Plot of Electron Number Density (Cant 45°, Array 45°)
y/z Planar Cut Through Center of Thruster
Log Base 10 of Normalized Units, 14 Contours from -3 to +6

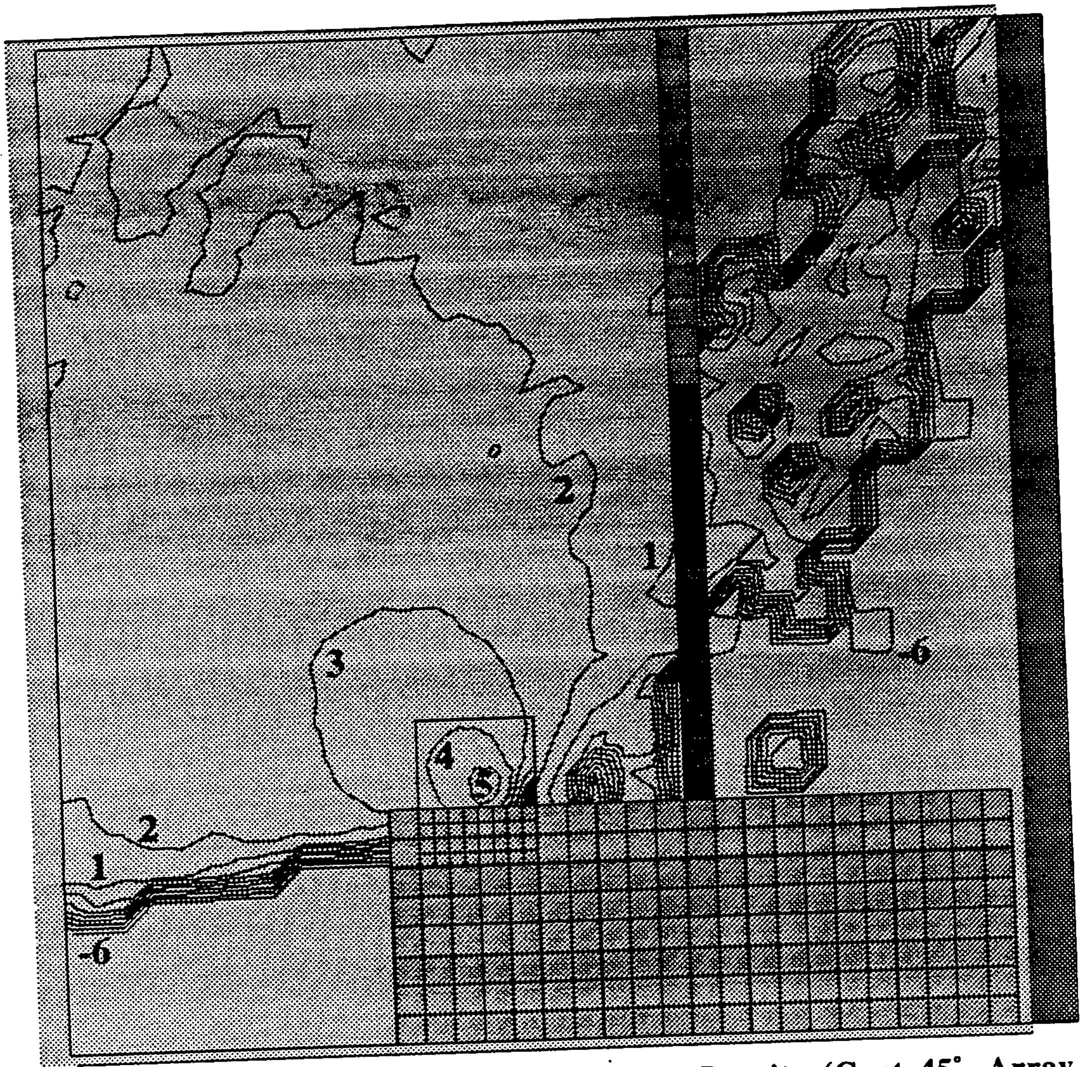


Figure B.3: Contour Plot of Neutral Number Density (Cant 45°, Array 45°)
y/z Planar Cut Through Center of Thruster
Log Base 10 of Normalized Units, 14 Contours from -3 to +6

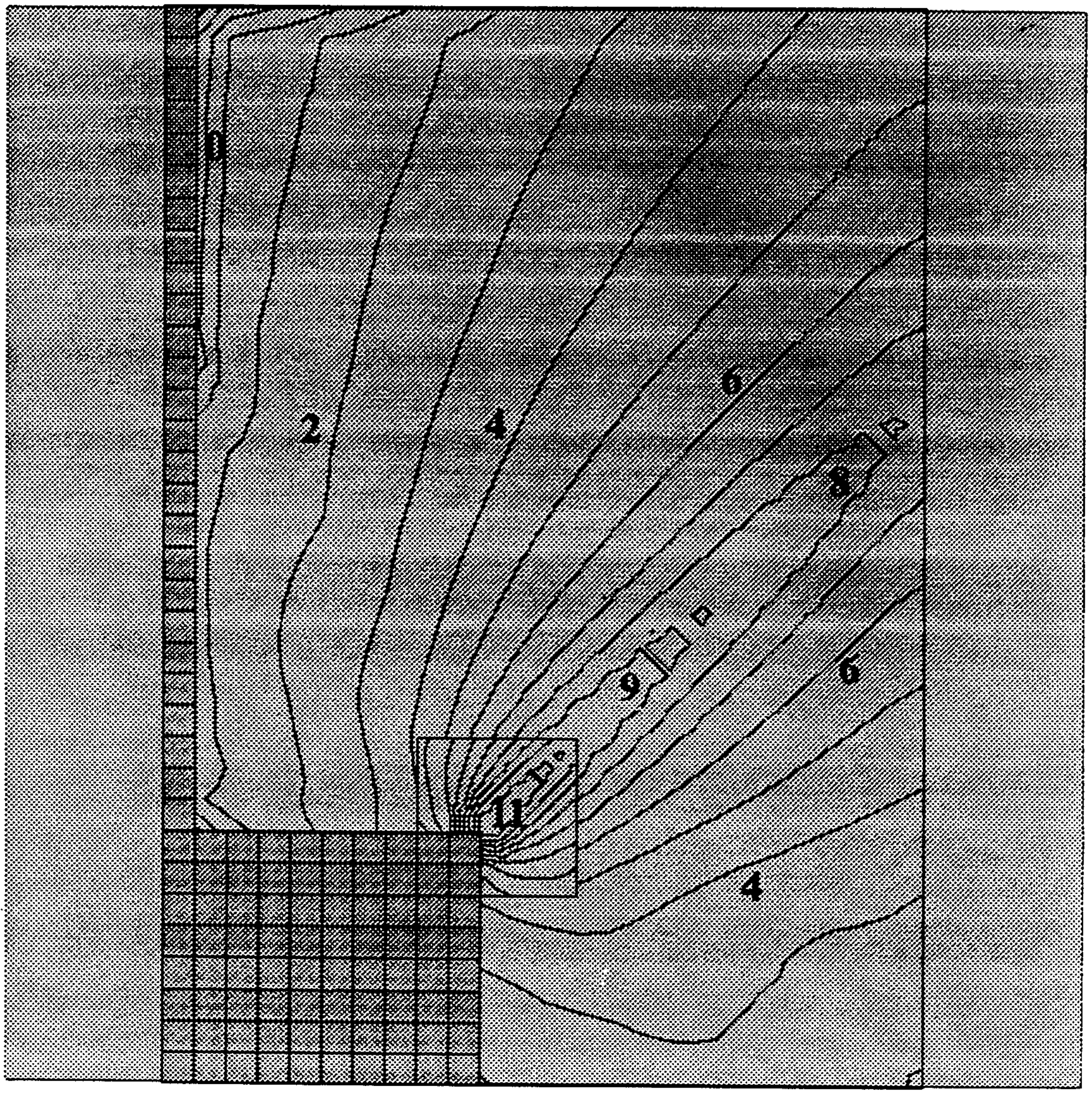


Figure B.4: Contour Plot of Potential (Cant 45°, Array 90°)
x/y Planar Cut Through Center of Thruster
Normalized Units, 20 Contours from -7 to +12

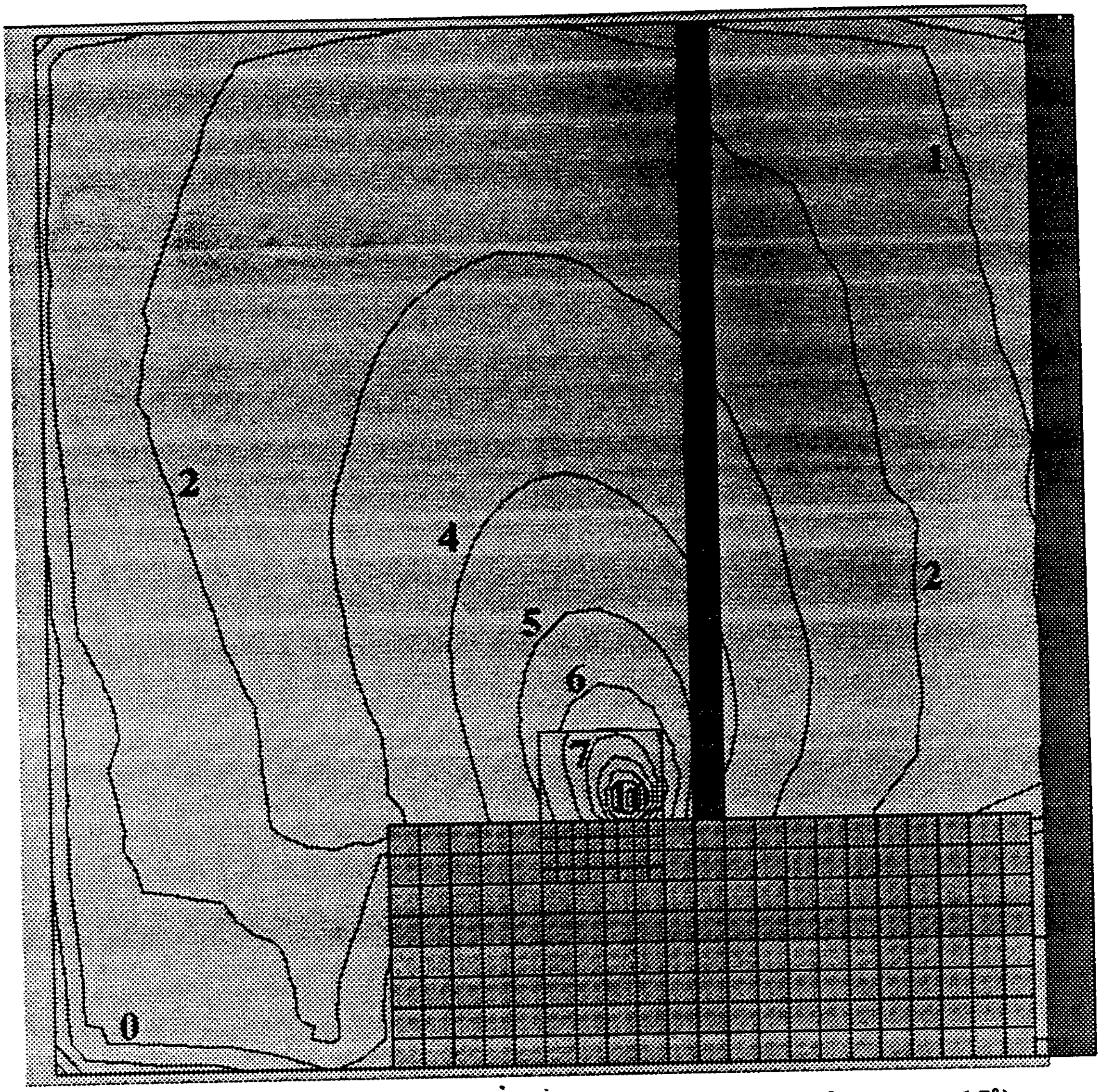


Figure B.5: Contour Plot of Potential (Cant 45°, Array 15°)
y/z Planar Cut Through Center of Thruster
Normalized Units, 20 Contours from -7 to +12

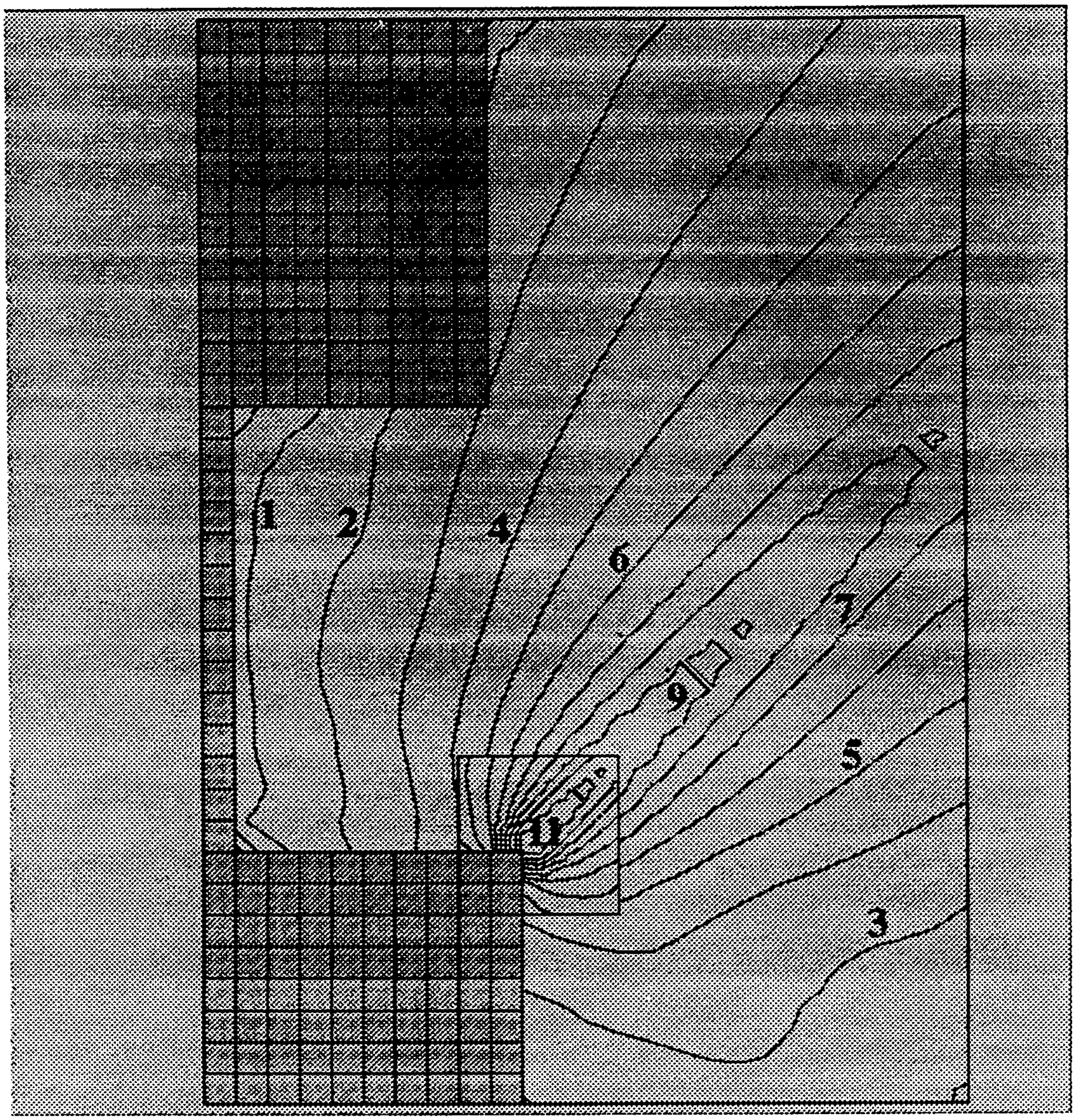


Figure B.6: Contour Plot of Potential (Cant 45°, Array 0°)
x/y Planar Cut Through Center of Thruster
Normalized Units, 20 Contours from -7 to +12

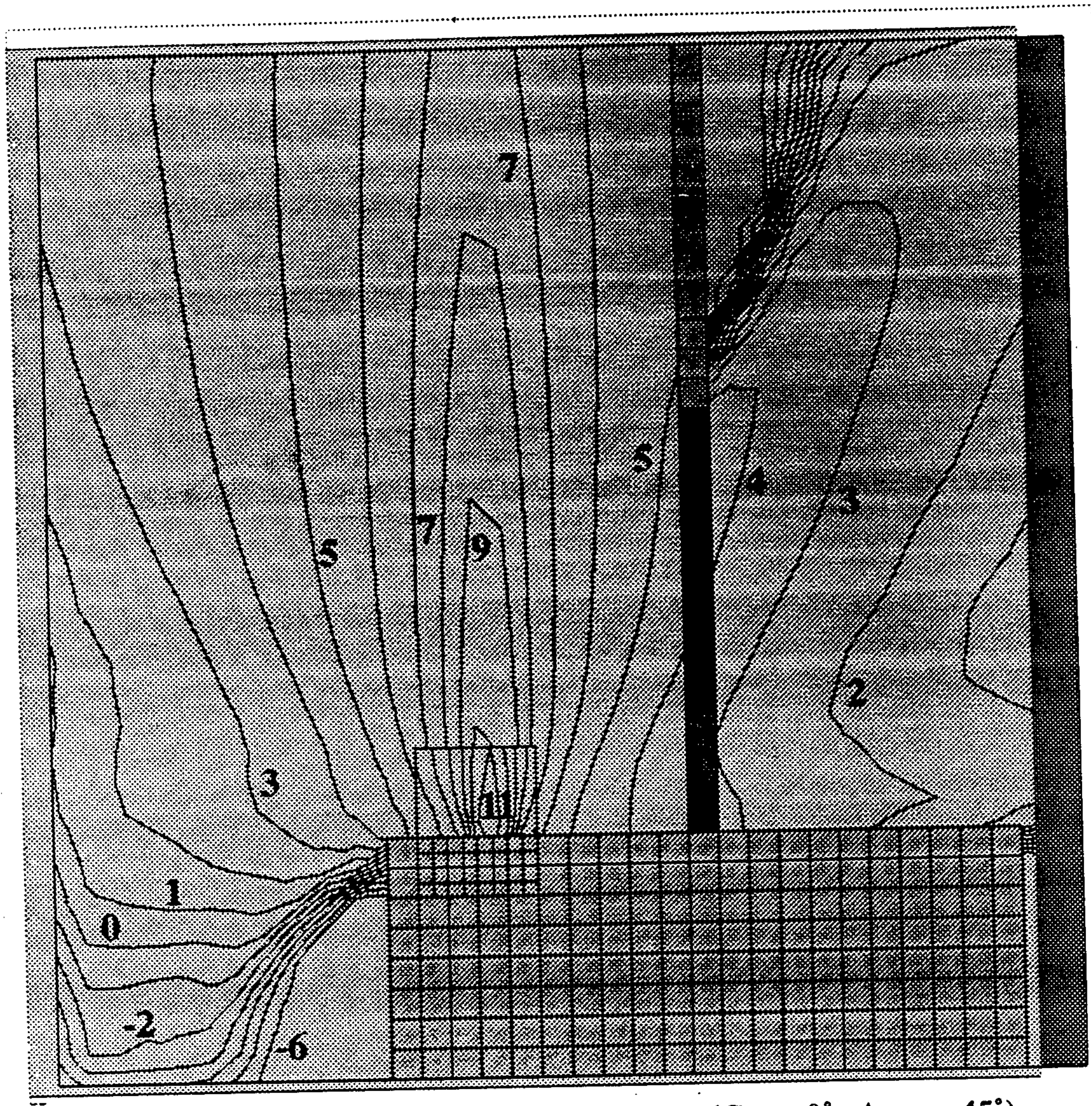


Figure B.7: Contour Plot of Potential (Cant 0°, Array 45°)
x/y Planar Cut Through Center of Thruster
Normalized Units, 20 Contours from -7 to +12

Bibliography

Note: The term "IEPC" refers to the International Electric Propulsion Conference.

Absalamov, S. K., Andreev, V.B. et al. "Measurement of Plasma Parameters in the Stationary Plasma Thruster (SPT-100) Plume and Its Effect on Spacecraft Components," AIAA Paper 92-3156, July 1992.

Albini, F.A. "Approximate Computation of Underexpanded Jet Structure," *AIAA Journal*, Vol. 3. No. 8, 1965, pp. 1535-1537.

Arkhipov, B.A. and Kozubsky, K.N. "The Development of the Cathodes-Compensators for Stationary Plasma Thrusters (SPT) in USSR," IEPC Paper 91-023, October 1991.

Arkhipov, B., Gnizdor, R. et al. "SPT-100 Module Lifetime Test Results," AIAA Paper 94-2854, June 1994.

Baganoff, D. and McDonald, J.D. "A Collision-Selection Rule for a Particle Simulation Method Suited to Vector Computers," *Phys. Fluids A*, Vol. 2, No. 7, July 1990, pp. 1248-1259.

Banks, P. "Collision Frequencies and Energy Transfer, Electrons," *Planet. Space Sci.*, Vol. 14, 1966, pp. 1085-1101.

Beattie, J.R. and Matossian, J.N. "High Power Ion Thruster Technology," NASA Report CR 187161, 1992.

Bird, G.A. "Breakdown of Translational and Rotational Equilibrium in Gaseous Expansions," *AIAA Journal*, Vol. 8, No. 11, 1970, pp. 1998-2003.

Bird, G.A. *Molecular Gas Dynamics,* 1st ed., Clarendon Press, Oxford, 1976.

Bird, G.A. "Breakdown of Continuum Flow in Free Jets and Rocket Plumes," *Proc. 12th Int. Symp. Rarefied Gas Dynamics, Prog. Astronaut. Aeronaut.*, Vol. 74, AIAA, New York, 1981, pp. 681-694.

Bird, G.A. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford University Press, Oxford, 1994.

Birdsall, C.K. "Particle-in-Cell Charged-Particle Simulations, Plus Monte Carlo Collisions With Neutral Atoms, PIC-MCC," *IEEE Transactions on Plasma Science*, Vol. 19, No. 2, April 1991, pp. 65-85.

Birdsall, C.K. and Langdon, A.B. *Plasma Physics Via Computer Simulation,* Adam Hilger, Bristol, 1991.

Bishaev, A.M. and Kim, V. "Local Plasma Properties in a Hall Current Accelerator with an Extended Acceleration Zone," *Sov. Phys. Tech. Phys.*, Vol. 23, No. 9, September 1978, pp. 1055-1057.

Bishaev, A.M., Kalashnikov, V.K. and Kim, V. "Numerical Modeling of Rarefied Plasma Plume Entering Neutral Environment Gas," IEPC Paper 93-137, September 1993.

Bogorad, A., Lichtin, D.A., Bowman, C., Armenti, J., Pencil, E. and Sarmiento, C. "The Effects of 1 kW Class Arcjet Thruster Plumes on Spacecraft Charging and Spacecraft Thermal Control Materials," *IEEE Transactions on Nuclear Science*, Vol. 39, No. 6, December 1992, pp. 1783-1789.

Borisov, B.S., Garkusha, V.I., Kozyrev, N.V., Korsun, A.G., Sokolov. L. Yu., and Strashinsky, V.A. "The Influence of Electric Thruster Plasma Plume on Downlink Communications in Space Experiments," AIAA Paper 91-2349, June 1991.

Borisov, B.S., Garkusha, V.I., Korsun, A.G., Sokolov, L. Yu, and Strashinski, V.A. "Exhaust Plasma Plume Impacts on Onboard Antenna Field Distribution," IEPC Paper 93-145, September 1993.

Boyd, I.D., "Modeling of Satellite Control Thruster Plumes," Ph.D. Thesis, University of Southampton, 1988.

Boyd, I.D., Cappelli, M.A. and Beattie, D.R. "Monte Carlo and Experimental Studies of Nozzle Flow in a Low-Power Hydrogen Arcjet," AIAA Paper 93-2529, June 1993.

Boynton, F.P. "Exhaust Plumes from Nozzles with Wall Boundary Layers," *J. Spacecraft Rockets*, Vol. 5, No. 10, 1968, pp. 1143-1147.

Bugrova, A.I., Kim, V., Maslennikov, N.A., and Morozov, A.I. "Physical Processes and Characteristics of Stationary Plasma Thrusters With Closed Electron Drift," IEPC Paper 91-079, October 1991.

Byers, David C. "Electron Bombardment Thruster Field and Particle Interfaces," *J. Spacecraft*, Vol. 16, No. 5, Sept.-Oct. 1979, pp. 289-301.

Carney, Lynnette M. "Evaluation of the Communications Impact of a Low Power Arcjet Thruster," AIAA Paper 88-3105, July 1988.

Carruth, M.R. and Brady, M.E. "Measurement of the Charge-Exchange Plasma Flow from an Ion Thruster," *J. Spacecraft*, Vol. 18, No. 5, Sept.-Oct 1981, pp. 457-461.

Carruth, M.R. "A Review of Studies on Ion Thruster Beam and Charge-Exchange Plasmas," AIAA Paper 82-1944, November 1982.

Chapman, B. *Glow Discharge Processes: Sputtering and Plasma Etching,* Wiley and Sons, New York, 1980.

Cho, M. and Hastings, D.E. "Dielectric Charging Processes and Arcing Rates of High Voltage Solar Arrays," *J. Spacecraft*, Vol. 26, No. 6, June 1991, pp. 698-706.

Crofton, M.W. "UK-10 Foreign Comparative Test Program Test and Evaluation Report," Aerospace Corporation Technical Report No. TOR-94(4507)-2, August 1995.

Crow, J.E., Auer, P.L. and Allen, J.E. "The Expansion of a Plasma into a Vacuum," *J. Plasma Physics*, Vol. 14, Pt. 1, pp. 65-76.

Cybulski, R.J., Shellhammer, D.M., Lovell, R.R., Domino, E. J. and Kotnik, J.T. "results from SERT I Ion Rocket Flight Test," NASA TN D-2718, 1965.

Dalgarno, A., McDowell, M.R.C. and Williams, A. "The Mobilities of Ions in Unlike Gases," *Philosopical Transactions of the Royal Society of London*, Vol. 250, No. A. 982, April 1958, pp. 411-425.

Dettleff, Georg. "Plume Flow and Plume Impingement in Space Technology," *Prog. Aerospace Sci.*, Vol. 28, No. 1, pp. 1-71, 1991.

Dickens, James C., Kristiansen, M. and O'Hair, E. "Plume Model of Hall Effect Plasma Thruster with Temporal Consideration," IEPC Paper 95-171, September 1995.

Elgin, J.B. "Getting the Good Bounce: Techniques for Efficient Monte Carlo Analysis of Complex Reacting Flows," Spectral Sciences Inc., SSI-TR-28, Burlington, MA. Date unknown.

Fearn, D.G., Martin, A.R. and Smith, P. "Ion Propulsion Development in the UK," AIAA Paper 93-2603, 1993.

Fife, John Michael. "Two-Dimensional Hybrid Particle-In-Cell Modeling of Hall Thrusters," Master's Thesis submitted to the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology, May 1995.

Gabdullin, F.F., Garkusha, V.I., Korsun, A.G., and Tverdokhlebova, E.M. "Radio Wave Refraction in Exhaust Plasma Plumes," IEPC Paper 93-144, September 1993.

Gavryushin, V.M. and Kim, V. "Effect of the Characteristics of a Magnetic Field on the Parameters of an Ion Current at the Output of an Accelerator with Closed Electron Drift," Sov. Phys. Tech. Phys., Vol. 26, No. 4, April 1981, pp. 505-507.

Gallimore, personal communication, 1996.

Garner, Charles E., Brophy, John R., Polk, James E., Pless, Lewis C. and Starling, Dan A. "A 5,730-Hr. Cyclic Endurance Test of the SPT-100," IEPC Paper 95-179, September 1995.

Hirakawa, M. and Arakawa, Y. "Numerical Simulation of Plasma Particle Behavior in a Hall Thruster," AIAA Paper 96-3195, July 1996.

Hill, J. A. F. and Draper, J.S. "Analytical Approximation for a Flow From a Nozzle into Vacuum," *Journal of Spacecraft and Rockets*, Vol. 3, No. 10, 1966, pp. 1552-1554.

Hoskins, W.A., Kull, A.E., and Butler, G.W. "Measurement of Population and Temperature Profiles from an Arcjet Plume," AIAA Paper 92-3240, July 1992.

Hockney, R.W. and Eastwood, J.W. *Computer Simulations Using Particles*, Adam Hilger, Bristol, 1988.

Kaufman, Harold R. "Charge-Exchange Plasma Generated by an Ion Thruster," NASA CR-134844, June 1975.

Kaufman, Harold R. "Technology of Closed-Drift Thrusters," AIAA Paper 83-1398, June 1983.

Kim, Sang-Wook, Foster, John E. and Gallimore, Alec D. "Very-Near-Field Plume Study of a 1.35 kW SPT-100," AIAA Paper 96-2972, July 1996.

King, Lyon V. and Gallimore, Alec D. "A Gridded Retarding Pressure Sensor for Ion and Neutral Particle Analysis in Flowing Plasmas," AIAA Paper 96-2983, July 1996.

King, Lyon B. and Gallimore, Alec D. "Ionic and Neutral Particle Transport Property Measurements in the Plume of an SPT-100," AIAA Paper 96-2712, July 1996.

Komatsu, G.K. and Sellen Jr., J.M. "Beam Efflux Measurements," NASA CR-135038, June 1976.

Komurasaki, Kimiya, Hirakawa, Miharu and Yoshihiro, Arakawa. "Plasma Acceleration Process in a Hall Current Thruster," IEPC Paper 91-078.

Larson, Wiley J. and Wertz, James R. *Space Mission Analysis and Design*, 2nd ed., Microcosm, Torrance, CA, 1992.

Lengrand, J.C. "Approximate Calculation of Rocket Plumes with Nozzle Boundary Layers and External Pressure," *Proc. 9th Int. Symp. Rarefied Gas Dynamics*, DFVLR Press, Porz-Wahn, 1974.

Lengrand, J.C., Allegre, J. and Raffin, M. "Experimental Investigation of Underexpanded Exhaust Plumes," *AIAA Journal*, Vol. 14, No. 5, pp. 692-694.

Lide, David R. (ed.), *CRC Handbook of Chemistry and Physics 75th ed.,* CRC Press, Boca Raton, 1994.

Liebeskind, J.G., Hanson, R.K. and Cappelli, M.A. "Plume Characteristics of an Arcjet Thruster," AIAA Paper 93-2530, June 1993.

Ling, Hao, Kim, Hyeongdong, Hallock, Gary A., Birkner, Bjorn W., and Zaman, Afroz, J.M. "Effect of an Arcjet Plume on Satellite Reflector Performance," *IEEE Transactions on Antennas and Propagation*, Vol. 39, No. 9, September 1991.

Lubachevsky, Boris D. "How to Simulate Billiards and Similar Systems," *Journal of Computational Physics*, Vol. 94, 1991. pp. 255-283.

Manzella, David H. "Stationary Plasma Thruster Plume Emissions," IEPC Paper 93-097, September 1993.

Manzella, David H. "Stationary Plasma Thruster Ion Velocity Distribution," AIAA Paper 94-3141, June 1994.

Manzella, David H. and Sankovic, John M. "Hall Thruster Ion Beam Characterization," AIAA Paper 95-2927, July 1995.

Marrese, Colleen, Gallimore, Alec D. et al. "An Investigation of Stationary Plasma Thruster Performance with Krypton Propellant," AIAA Paper 95-2932, July 1995.

Marrese, Colleen M., Haas, James M., Domonkos, Matthew T., Gallimore, Alec D., Tverdokhlebov, Sergei and Garner, Charles E. "D-100 Performance and Plume Characterization," AIAA Paper 96-2969, July 1995.

Mitchner, M. and Kruger, C. *Partially Ionized Gases,* Wiley-Interscience, 1973.

Monheiser, J.M. "Advanced Electric Propulsion Research - 1990," NASA Report CR-187103, 1991.

Myers, Roger M. and Manzella, David H. "Stationary Plasma Thruster Plume Characteristics," IEPC Paper 93-096, September 1993.

Ohler, S., Gilchrist, B. and Gallimore, A. "Microwave Plume Measurements of an SPT-100 Using Xenon and a Laboratory Model SPT Using Krypton," AIAA Paper 95-2931, July 1995.

Ohler, Shawn G., Ruffin, Alranzo B., Gilchrist, Brian E. and Gallimore, Alec D. "RF Signal Impact Study of an SPT," AIAA Paper 96-2706, July 1996.

Olsen, R.C., Whipple, E.C. and Purvis, C.K. "Active Modification of ATS-5 and ATS-6," *Spacecraft Potentials, Preprint of Proceedings , Symposium on the Effect of the Ionosphere on Space and Terrestrial Systems*, IES, Paper 4-9, Jan. 1978.

Pencil, Eric J. "Preliminary Far-Field Plume Sputtering Characterization of the Stationary Plasma Thruster (SPT-100)," IEPC Paper 93-098 (also NASA Technical Memorandum 106464), September 1993.

Pencil, Eric J., Randolph, Thomas, and Manzella, David H. "End-of-life Stationary Plasma Thruster Far-field Plume Characterization," AIAA Paper 96-2709, July 1996.

Peng, X., Ruyth, W. and Keefer, D. "3D Particle Simulation of Grid Erosion in Ion Thrusters," AIAA Paper 91-119, 1991.

Peng, X., Ruyten, W. and Keefer, D. "Comparison of 2D and 3D Models of Grid Erosion in an Ion Thruster," AIAA Paper 91-2120, 1991.

Peng, X., Ruyten, W. and Keefer, D. "Further Study of Effect of the Downstream Plasma Condition on Accelerator Grid Erosion in an Ion Thruster," AIAA Paper 92-3829, 1992.

Pollard, J.E. "Plume Measurement with the T5 Xenon Ion Thruster," Aerospace Corporation Report No. TR-94(4507)-2, 1994.

Press, William H., Vetterling, William T., Teukolsky, Saul A., and Flannery, Brian P. *Numerical Recipes in C: The Art of Scientific Computing*, 2nd. ed., Cambridge Univ. Press, Cambridge, 1992.

Randolph, T., Pencil, E. and Manzella, D. "Far-Field Plume Contamination and Sputtering of the Stationary Plasma Thruster," AIAA Paper 94-2855, June 1994.

Randolph, T. Space Systems/Loral. Internal Memorandum, 1994.

Rapp, D. and Francis, W.E. "Charge Exchange Between Gaseous Ions and Atoms," *Journal of Chemical Physics*, Vol. 37, No. 11, 1962, pp. 2631-2645.

Rhee, M.S. and Lewis, M.J. "Numerical Simulation of Stationary Plasma Thruster Exhaust Plume," AIAA Paper 95-2928, July 1995.

Roberts, L. and South, J.C. "Comments on Exhaust Flow Field and Surface Impingement," *AIAA Journal*, Vol. 2, No. 5, 1964, pp. 971-973.

Robinson, R.S. , Kaufman, H.R. and Winder, D.R. "Plasma Propagation Simulation Near an Electrically Propelled Spacecraft," *J. Spacecraft*, Vol. 19, No. 5, Sept.-Oct. 1982, pp. 445-450.

Rosenberg, D. and Wehner, G.K. "Sputtering Yields for Low Energy He+-, Kr+-, and Xe+- Ion Bombardment," *Journal of Applied Physics*, Vol. 33, No. 5, May 1962, pp. 1842-1845.

Roux, J.A. and McCay, T.D. *Spacecraft Contamination: Sources and Prevention*. Progress in Astronautics and Aeronautics Series, Vol. 91, AIAA, Washington D.C., 1984.

Ruyten, W.M. "Density-Conserving Space Factors for Particle Simulations in Cylindrical and Spherical Coordinates," *J. Comp. Phys.*, Vol. 105, 1993, pp. 224-232.

Ruyten, W.M. "Spectroscopic Investigation of a Low-Power Arcjet Plume," AIAA Paper 93-1790, June 1993.

Sack, Ch. and Schamel, H. "Evolution of a Plasma Expanding into Vacuum," *Plasma Physics and Controlled Fusion*, Vol. 27, No. 7, 1985, pp. 717-549.

Samanta Roy, Robie I. "Numerical Simulation of Ion Thruster Plume Backflow for Spacecraft Contamination Assessment," MIT Department of Aeronautics and Astronautics, Ph.D. Thesis, Cambridge, MA, June 1995.

Samanta Roy, R. I., Hastings, D.E. and Gatsonis, N.A. "A Review of Contamination from Electric Propulsion Thrusters," AIAA Paper 94-2469, June 1994.

Sellen Jr., J.M. et al., "Generation and Diagnosis of Synthesized Plasma Streams," *Rev. Sci. Instr.*, Vol. 36, March 1965, pp.316-322.

Sibulkin, M. and Gallaher, W.H. "Far Field Approximation for a Nozzle Exhausting into a Vacuum," *AIAA Journal*, Vol. 1, No. 6, 1963, 1452-1453.

Simons, G.A. "Effect of Nozzle Boundary Layers on Rocket Exhaust Plumes," *AIAA Journal*, Vol. 10, No. 11, 1972, pp. 1534-1535.

Staggs, J.F., Gula, E.P. and Kerslake, W.R. "Distribution of Neutral Atoms and Charge-Exchange Ions Downstream of an Ion Thruster," *J. Spacecraft*, Vol. 5, No. 2, Feb. 1968, pp. 159-164.

Staskus, J.V. and Burns, R.J. "Deposition of Ion Thruster Effluents on SERT II Spacecraft Surfaces," NASA TM X-2084, 1970.

Takegahara, H., Kasai, Y., Gotoh, Y., Miyazaki, K., Hayakawa, Y., Kitamura, S. Nagano, H. and Nakamaru, K. "Beam Characteristics Evaluation of ETS-VI Xenon Ion Thruster," IEPC Paper 93-325, 1993.

Wang, Joe. "Electrodynamic Interactions Between Charged Space Systems and the Ionospheric Plasma Environment," MIT Department of Aeronautics and Astronautics, Ph.D. Thesis, Cambridge, MA, June 1991.

Wang, J., Brophy, J., Liewer, P. and Murphy, G. "Modeling Ion Thruster Plumes," AIAA Paper 95-0596, January 1995.

Williams, John D. and Wilbur, Paul J. "Experimental Study of Plasma Contactor Phenomena," J. Spacecraft, Vol. 27, No. 4, 1990, pp. 634-641.

Worlock, R.M., James, E.L., Hunter, R.F. and Bartlett, R.O. "The Cesium Bombardment Engine North-South Stationkeeping Experiment on ATS-6," AIAA Paper 75-363, March 1975.

IEPC- 95-160

Axisymmetric PIC-DSMC Simulations of SPT Plumes

D. Oh and D. Hastings
Space Systems Laboratory
Massachusetts Institute of Technology
Cambridge, MA, USA

24th International Electric Propulsion Conference

"Leningradsky 55", Congress Center
Moscow, Russia, September 19-23, 1995

AXISYMMETRIC PIC-DSMC SIMULATIONS OF SPT PLUMES

DAVID Y. OH*
DANIEL HASTINGS*

Abstract

A computational model of a Stationary Plasma Thruster (SPT) plume has been constructing using a quasi-neutral Particle-in-Cell/Direct Simulation Monte Carlo (PIC-DSMC) model. This model is based on theoretical work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. The resulting axisymmetric simulation and its underlying assumptions are described in detail, and comparisons are made between the computational model and existing experimental measurements of the ion current density in the plume region. The comparison highlights weaknesses in both the model and the data. The simulated and experimental results agree well when two extra current sources are included in the computational model. These sources represent a lack of Xe^{++} in the simulation and the presence of external current sources in the experimental tests. Although further work is needed, the results suggest that the simulation is accurately modeling the physics of the plume region and that it can give predictions of backflow currents under operational conditions. Further work is planned to include Xe^{++} in future simulations.

Nomenclature

A	= Thruster Exit Area
B	= Magnetic Field Strength
E	= Electric Field
E_r	= Radial Electric Field
E_z	= Axial Electric Field
W_{ion}	= Ion Macroparticle Weight
$W_{neutral}$	= Neutral Macroparticle Weight
T_e	= Electron Temperature
T_i	= Ion Temperature
Z_i	= Ionization Number
a_o	= Speed of Sound
c_r	= Relative Speed between Collision Partners
e	= Elementary Charge = 1.6×10^{-19} C
k	= Boltzman's Constant = 1.38×10^{-23} J/K
m_e	= Electron mass = 9.11×10^{-31} kg.
m_i	= Xe ion mass = 2.18×10^{-25} kg.
$\dot{m}$	= Anode Propellant Mass Flow Rate
n_e	= Electron Number Density
n_i	= Ion Number Density
n_o	= Neutral Number Density
n_{ref}	= Reference Electron Number Density
p	= Pressure
r	= Radial Position
r_a	= Insulator Ring Width
t_{global}	= Global Time Count
t_{local}	= Local Time Count
$v_{\perp}$	= Velocity Perpendicular to B field
v_z	= Axial Ion Velocity
u	= Velocity
z	= Axial Position
β	= Random Number Between 0 and 1
λ_d	= Debye Length
λ_{en}	= Electron-Neutral Mean Free Path
λ_{ee}	= Electron-Electron Mean Free Path
λ_{ei}	= Electron-Ion Mean Free Path
ε_o	= 8.85×10^{-12} F/m
φ	= Electrical Potential
η_i	= Ionization Fraction
ρ_{ge}	= Electron Gyro Radius
ρ_{gi}	= Ion Gyro Radius
σ	= Collision Cross Section
σ_{CEX}	= Charge Exchange Collision Cross Section
$\overline{\sigma}_{ee}$	= Mean Electron-Self Collision Cross Section
$\overline{\sigma}_{ie}$	= Mean Ion-Electron Collision Cross Section
θ	= Angular Position
$\ln \Lambda$	= Spitzer Logarithm

* Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA

Introduction

When compared to chemical propulsion systems, electric propulsion offers substantial fuel savings for stationkeeping applications. One system which has shown particular promise is the Stationary Plasma Thruster (SPT), also referred to as the Hall Thruster. These devices have a near optimum specific impulse for stationkeeping and one model, the SPT-100, is being actively marketed by Russian companies. Western designers have expressed concern that the SPT's plasma plume may erode and contaminate sensitive surfaces, cause parasitic current losses, and interfere with communications signals. These concerns must be addressed before SPT's can be used for commercial applications.

A critical step in addressing these issues is defining and modeling the characteristics of an SPT plume. Although much experimental work has been conducted on SPT thrusters,[1-5] only limited efforts have been made to model the processes occurring in the plume region.[6] In addition, the experiments to date have been conducted in vacuum tanks, and may not accurately simulate on-orbit conditions. More detailed and realistic models are needed to fully characterize the plume region and to understand the relationship between ground based experimental data and actual operational conditions.

One issue of major concern to spacecraft designers is the effect of charge exchange (CEX) collisions on ions in the plume region. A CEX collision occurs when a "slow" neutral and a "fast" ion exchange an electron to create a fast neutral and a slow ion. The resulting slow ion can be accelerated back towards the spacecraft, creating backflow and impingement problems. This paper describes a detailed computational model of an SPT's plume region. The model uses a combination of particle-in-cell (PIC) and direct simulation Monte Carlo (DSMC) methods to model CEX collision and track ions and neutrals in the plume region. The model runs quickly, compares favorably to experimental data and can be used to model the plume on meter length scales. This paper describes the model in detail, presents results for an SPT-100 operating in an axisymmetric geometry, and compares the results to existing experimental data. The paper also presents simulations of the SPT-100 operating under conditions which have not been studied in ground based tests. This paper concentrates on modeling of the SPT-100 because this is most extensively studied thruster at this time. The model can be modified and applied to other Hall thrusters as they are developed.

Section 1 presents the theoretical basis for our computational model. Section 2 describes the PIC-DSMC algorithm in detail and describes some of the tests used to verify the model. Section 3 presents simulated results and compares them to existing experimental data. Section 4 presents conclusions and discusses plans for future work.

Section 1: Basic Theory and Fundamental Parameters

The SPT-100 thruster has been studied extensively in ground tests and many of its basic characteristics are well documented. Table 1 summarizes some relevant results.

Inner Insulator Diameter[3]	56 mm
Outer Insulator Diameter[3]	100 mm
Propellant	Xe
Propellant Flow Rate[1]	5.0-5.2 mg./sec.
Fraction of Propellant Directed to Cathode[1]	~10%
Electron Temperature[5] (T_e)	2-4 eV
Axial Ion Velocity[3] (v_z)	~16000 m/s
Fraction of Propellant Ionized in Discharge Chamber [1] (η_i)	>95%

Table 1: SPT-100 Basic Characteristics

The characteristics of the plume region can be estimated from Table 1 based on some simple assumptions. The approximate ion and neutral densities at the exit are given by

$$n_i = \eta_i \dot{m} / v_z A$$

$$n_o = (1 - \eta_i) \dot{m} / a_o A \qquad (1)$$

Substituting values from Table 1 gives exit densities of $n_i = 2.4 \times 10^{17}$ m^{-3} and $n_o = 9.3 \times 10^{17}$ m^{-3}. The plasma and neutral densities downstream of the thruster can be estimated by assuming that the density falls as $(r_a/z)^2$, and results of this calculation are given in Table 2. The resulting plasma densities are consistent with existing experimental data.[5] No experimental measurements are presently available for the neutral density.

Once the ion and neutral density are known, a variety of fundamental parameters can be calculated. Table 2 lists many of these parameters. The methods used to obtains these values and their implications for the computation model are discussed in the rest of this section.

	Radial Position	
	0.2 m	**1.0 m**
n_i/n_e	2.88×10^{15} m^{-3}	1.15×10^{14} m^{-3}
n_o	1.12×10^{16} m^{-3}	4.49×10^{14} m^{-3}
B	~1 G	~0.01 G
λ_d	0.03 cm	0.14 cm
ρ_{ge}	6.7 cm	670 cm
ρ_{gi}	230 m	>20000 m
λ_{en}	85 m	340 m
λ_{ee}	130 m	3200 m
λ_{ei}	270 m	6700 m

Table 2: Estimates of Fundamental Parameters in the Plume Region

Debye Length

The characteristic length over which charges are neutralized in a plasma is given by the Debye length

$$\lambda_d = \sqrt{\varepsilon_o k T_e / n_e e^2}$$

The Debye lengths shown in Table 2 are based on an electron temperature of 4 eV. These relatively small values indicate that the Debye length is small with respect to features of interest in the plume region. Even at z = 4 m, the Debye length is only 0.55 cm. It can therefore be assumed that the plume is quasi-neutral everywhere except in a thin sheath region near solid surfaces. We use this assumption to simplify our PIC model and avoid the direct solution of Poisson's equation. The details of our method are discussed in Section 2.

Gyro Radius

A magnetic field is present in an SPT's anode region to capture electrons and encourage impact ionization of the Xe propellant. The extent to which this magnetic field "leaks" into the plume region has been measured experimentally, and is shown in Figure 1.[7] The importance of the magnetic field in the plume region is measured by the gyro radius, which is given by

$$\rho_{gi/e} = m_{i/e} v_{\perp} / eB$$

The results shown in Table 2 demonstrate that the plume ions are effectively unmagnetized. In addition, although electrons are strongly magnetized inside the discharge chamber, the electron gyro radius becomes quite large 20 cm from the thruster and extremely large 1 m away. As a result, the magnetic field can be ignored at distances > 25 cm from the thruster exit. We note also that recent experimental work suggests that plume characteristics are only indirectly affected by the magnetic field. The ion distribution appears to be determined well inside the thruster, and varying the strength of the magnetic field has no measurable influence on plume.[4] Since our simulation begins at the thruster exit, we choose to neglect the magnetic field throughout the plume region.

Mean Free Path

The importance of collisions is measured by the mean free path, which is given by the well known relationship

$$\lambda = 1/(n\sigma) \qquad (2)$$

In an SPT plume, electrons can collide with three different species: ions, neutrals, and other electrons. Each species has its own collision cross section and mean free path. The total electron-Xe (neutral) collision cross section has been determined experimentally. For electrons with an average velocity of 4 eV, the cross section is ~2.6×10^{-19} m^2.[8] The resulting λ_{en}'s are given in Table 2. These values are large with respect to the plume, indicating that if the electrons are unmagnetized, electron-neutral collisions can be neglected.

The electron-electron and electron-ion collision cross sections are based on coulomb interactions and can be calculated from theory. Banks has derived the average electron-ion cross section for $T_e/m_e >> T_i/m_i$.[9] The result, in cgs units, is

$$\overline{\sigma}_{ie} = \frac{\pi}{2}\frac{\left(Z_i e^2\right)^2 \ln\Lambda}{\left(kT_e\right)^2}$$

Table 2 gives the resulting λ_{ie} with $T_e = 4$ eV. These values are also extremely large when compared to features of interest.

The mean self collision cross section for singly ionized species is given by[10]

$$\overline{\sigma}_{ee} \cong \left(5.85\times10^{-10}\right)\ln\Lambda / T_e^2 \text{ m}^2$$

For an electron temperature of 4 eV, this gives $\sigma = 2.7 \times 10^{-18}$ m^2, which results in the λ_{ee}'s listed in Table 2. These values are also much larger than features of interest in the plume. Therefore, in the absence of a magnetic field, electrons are effectively collisionless in the plume region.

One additional process of interest is charge exchange collisions. The cross section for CEX for singly ionized Xenon is given by[11]

$$\sigma_{CEX} = \left(k_1 \ln c_r + k_2\right)^2 \cdot 10^{-20} m^2 \qquad (3)$$

where $k_1 = -0.8821$, $k_2 = 15.1262$ and c_r is in m/s. Although the resulting mean free path is large, the resulting backflow is non-negligible and is included in our computational model.

In summary, the results shown in Table 2 indicate that the bulk of the SPT-100 plume consists of a quasi-neutral, unmagnetized plasma in which the electrons are effectively collisionless. In addition, although there is a small magnetized region in front of the thruster ($z < 0.25$ m), experimental work suggests that the overall plume structure may be insensitive to changes in the strength of the magnetic field. We therefore choose to neglect the magnetic field throughout the plume region. Charge exchange collisions are also of importance in the plume region. Although the CEX mean free path is long, the resulting backflow is non-negligible and should be included in any plume model.

Section 2: Computational Model

In order to model the SPT-100 plume, we constructed a PIC-DSMC simulation of the plume region. Our model is a combination of the PIC and DSMC methods as described by Birdsall[12] and Bird[13] respectively. Since these methods are well known individually, only features specific to this model will be described in detail. A flowchart of our basic method is shown in Figure 2.

The plume simulation moves both ions and neutrals by integrating the particle equations of motion using the leapfrog method. In axisymmetric coordinates, the equations of motion (without collisions) are

$$\ddot{r} - r\dot{\theta} = qE_r / m_i \qquad r\ddot{\theta} + 2\dot{r}\dot{\theta} = 0 \qquad \ddot{z} = qE_z / m_i$$

The electric field is determined by differentiating the potential. In conventional PIC schemes, the potential is obtained by solving Poisson's equation. In a quasi-neutral plasma, however, the potential can be obtained by inverting the electron momentum equation. For collisionless, unmagnetized electrons, the momentum equation is

$$m_e n_e (\mathbf{u}\cdot\nabla)\mathbf{u} = e n_e \mathbf{E} - \nabla p$$

At low drift velocities, the first (inertial) term can be neglected. We also assume that the electrons are isothermal, resulting in the familiar Boltzman relationship.

$$n_e = n_{ref}\exp\left(e\phi / kT_e\right) \qquad (4)$$

The potential is obtained using the following procedure. The ion density is determined by weighting macroparticles to the nodes of an embedded grid. The weighting functions for an axisymmetric geometry were worked out by Ruytan, and are given by [14]

$$S_j = \frac{\left(r_{j+1} - r\right)\left(2r_{j+1} + 3r_j - r\right)}{2\left(r_{j+1}^2 - r_j^2\right)} \qquad S_{j+1} = \frac{\left(r - r_j\right)\left(3r_{j+1} + 2r_j - r\right)}{2\left(r_{j+1}^2 - r_j^2\right)}$$

Where S is the weighting factor and subscripts refer to grid coordinates. Since the plume is quasi-neutral, the electron density can be set equal to the ion density. Equation (4) can then inverted to obtain the potential, thus avoiding the direct solution of Poisson's equation. The result is a fast method which is free of many of the restrictions present on conventional PIC codes. Because the plasma is assumed to be quasi-neutral, the Debye length does not limit the size of the grid cells. Instead, grid cell sizes are limited by geometry and the need to resolve potential gradients. Similarly, time steps are limited by energy conservation and the need to ensure that particles move a small distance relative to the width of local potential gradients. In practice, we insure that the fastest beam ions in the simulation move no more than one grid cell in each time step.

CEX collisions are modeled between move steps using a modified DSMC method which is outlined in Figure 4. A conventional selection-rejection scheme is used to determine collision pairs and the local time counter method is used to determine collision frequencies. When a collision takes place between two particles with equal weights, the ion and neutral particle velocities are exchanged, thus creating a "slow" ion and a "fast" neutral. If the simulated neutrals have a larger macroparticle weight than the simulated ions, an additional Monte Carlo selection is made to determine whether the neutral velocity should be modified. This ensures that the total energy and momentum in the simulation are statistically conserved over many collisions. In the limit as the weights become equal, our scheme becomes a conventional DSMC method. In the other limit, as the neutral weight becomes infinitely large, our scheme approaches the conditions simulated in a PIC-MCC method. We use equal weights when simulating an SPT-100 operating in vacuum and unequal weights when simulating ground tests. In all cases, all members of one species (either ions or neutrals) have the same macroparticle weight. The same grid is used for both the PIC and DSMC parts of the simulation.

The accuracy of our DSMC algorithm was tested by simulating a collimated beam of ions entering a stationary neutral gas and comparing the results to theory. The simulated collision frequency was shown to be accurate as long as more than twenty macro particles (ten from each species) were present in the cell. The results shown in this paper are all produced with at least forty macro particles (twenty from each species) present in each cell.

Particles are loaded into the simulation at each time step to simulate the exit flow from an SPT-100. The number of ions and neutrals introduced in each time step is determined using equation (1) and assumes that the flow is 95% ionized. The ion distribution is determined from an empirical model developed from experimental measurements of the ion current density 4 mm from the thruster exit.[15] These measurements give the magnitude and direction of the ion current as a function of radial position at an unspecified propellant flow rate. We found that the measured current density can be described by the following functions

$$\alpha = 1730 - 2.30\times10^{5} r + 1.06\times10^{7} r^{2} - 2.05\times10^{8} r^{3} + 1.45\times10^{9} r^{4} \qquad (5)$$

$$j = -1210 + 8.40\times10^{4} r - 1.78\times10^{6} r^{2} + 1.18\times10^{7} r^{3} \qquad (6)$$

Where α is the divergence angle in degrees, j is the current density in mA/cm^2 and r is the radial position in meters. A model for the ion distribution is derived by assuming that ions leave the thruster with a drift velocity of 16000 m/s in the r/z plane. Variations in current density then correspond to variations in number density. Equation (6) can be normalized by the integrated beam current to give

$$P = 2.55 - 1.67\times10^{4} r^{2} + 7.71\times10^{5} r^{3} - 1.23\times10^{7} r^{4} + 6.50\times10^{7} r^{5} \qquad (7)$$

Equation (7) gives the probability that an ion crossing the exit plane has a radial position less than r. The following procedure can therefore be used to load plume ions. A random number P is chosen between 0 and 1. Equation (7) is then inverted computationally to give the particle's radial position. Equation (5) can then used to determine the particle's divergence angle. The divergence angle, in turn, can be used to calculate the axial and radial drift velocities. Finally, thermal components are added in the axial and radial direction based on temperatures of 3.4 eV and 8000 K respectively. The axial temperature matches measurements made by Manzella,[3] and the radial velocity is a reasonable guess. The particle's tangential velocity is determined independently by assuming a drift velocity of 250 m/s and a temperature of 800 K. These values also match measurements made by Manzella[3].

Because the cathode can not be directly included in an axisymmetric geometry, the propellant which normally flows through the cathode was assumed to flow through the anode as unionized propellant. The neutral distribution has not been measured in the plume region, so we assume that the neutrals have a temperature of 1000 K and are choked at the thruster exit. In order to verify the accuracy of our model, the kinetic and potential energy in the simulation were tracked and it was shown that energy was properly conserved in steady state. The final simulation was written in C and runs were carried out on a Digital Equipment Alpha 200 Workstation with 64 megabytes of RAM. Run times were typically between 2 and 4 hours.

Section 3: Results and Discussion

A series of simulations were run to examine the effect that the background pressure present in ground test facilities has on the structure of the SPT-100 plume. Four different ambient pressures were simulated as well as operation in an ideal vacuum. A single computational domain was used for all simulations and is shown in Figure 3. The presence of background gas was simulated by adding neutral Xe particles in a Maxwellian distribution and by including thermal Xe fluxes along boundaries with the ambient plasma. The background ion temperature was assumed to be 300 K and the number density was determined using the ideal gas law. Typical results are shown in Figures 5 and 6. The ambient pressure is 5.6 x 10^{-6} Torr and the anode propellant flow rate is 4.9 mg/sec.

Figure 5 shows a contour plot of the potential overlaid with vectors showing ion current direction and magnitude. Larger vectors correspond to the higher current magnitudes. The primary beam is clearly visible in Figure 5 as a region of high potential close to the r=0 axis. In addition to the main beam, however, a bump is visible in the potential to the front and side of the thruster and backflow current is clearly visible to the side and back of the thruster. The bump and the corresponding current flow is caused by CEX ions. Figure 6 shows a phase space plot of the simulated data with different species labeled by type. CEX ions are clearly visible in this diagram as a wing of low velocity ions which are accelerated sideways and backwards by the electric field. Xe neutrals are also visible in Figure 6 as a cluster of low speed particles near the origin. Although CEX ions begin with a relatively low speed, the plume's potential structure tends to drive these ions sideways and backwards, towards the spacecraft rather than away from it. These CEX ions can then impact spacecraft surfaces, causing contamination and erosion damage.

Figure 7 shows simulated ion current densities as measured at five different ambient pressures in an arc 60 cm from the thruster exit. The current was measured at angles from 0 to 100 degrees from the centerline and then reflected to obtain measurements at negative angles. As one would expect, higher ambient pressures result in higher backflow currents due to a higher CEX collision rate. Even when the thruster operates in full vacuum, however, backflow still occurs due to CEX collisions with unionized propellant. It should be noted that even at fairly lower pressure, the presence of ambient neutrals has a measurable effect on the backflow current. Our model provides a means of estimating the actual backflow which will be present under orbital conditions. Figure 7 indicates that the backflow current present in orbit can be much lower than that measured in ground tests.

In order to assess the accuracy of our PIC-DMSC model, the results shown in Figure 7 were compared to existing experimental data. Manzella measured the ion current density in an arc 60 cm from an SPT-100 thruster at angles from -100 to 100 degrees from the centerline and showed that the current density varies significantly with the facility pressure.[4] His results are shown in Figure 9, and direct comparisons to simulated data are shown in Figure 8. The peaks are not aligned in these plots because the experimental data is not exactly centered on zero. Although the same basic trends are present in both sets of data, there are notable differences in the shape and magnitude of these results. In particular, the computed results consistently underpredict the current density at all pressures and angles. As a result, the total beam current is much larger in the experimental case than it is in the simulations. The beam current can be calculated by numerically integrating the data presented in Figure 9. The results of such an integration are shown in Table 3. Two additional values are also shown in Table 3. The "expected value," is the beam current which would be present if the entire propellant flow were singly ionized as it passed though the anode. This approximately equals the integrated current measured in the simulated data. The "discharge current" is the experimentally measured current flow between the anode and cathode.

Pressure (Torr)	Current (A)
2.2 x 10^{-6}	4.69
5.6 x 10^{-6}	5.26
2.5 x 10^{-5}	5.97
6.3 x 10^{-5}	6.15
Expected Value	3.66
Discharge Current	~4.5

Table 3: Integrated Beam Currents

Table 3 shows that the experimentally measured beam current exceeds the expected current by between 28% and 68%. The amount by which the current exceeds the expected value varies with pressure, suggesting that some interaction is occurring between the thruster and the ambient background. In addition, Table 3 also shows that the beam current exceeds the discharge current. Since the beam current should always be less than the discharge current, this indicates that an external current source must be present in the experimental data. To determine the impact of this external current on our results, we conducted simulations in which the anode ion flow was increased by the difference between the expected value and the beam currents given in Table 3. The results of these simulations are shown in Figure 10.

Figure 10 shows that there is good agreement between the PIC-DSMC model and experimental data when the extra current shown in Table 3 is included in the simulation. The fraction of this extra current which is less than the discharge current is thought to be due to the presence of doubly ionized Xe (Xe^{++}), which is not presently included in the simulation. Experimental work suggests that Xe^{++} composes perhaps 20% of the plume,[2] which matches the difference between the expected and discharge current values. Work is currently underway to include Xe^{++} in the simulation. At higher pressures, however, a substantial fraction of the extra current exceeds the discharge current. We therefore conclude that an anomalous external current source is biasing the experimental data. The strength of this source is heavily influenced by the presence of ambient neutrals. This suggests that the current may be caused by an ambient plasma, either due to thermal currents or due to recirculation within the vacuum chamber. Further work is needed to clarify this issue.

In summary, the PIC-DSMC model presented in this paper compares fairly well to existing experimental data when the simulation is modified to include all current sources present in ground based experiments. The modified ion beam currents result in part from weaknesses in the model and in part from unknown current sources present in this particular experiment. When this additional current is included in the simulation, the simulated current density agrees quite well with experimental measurements. The results presented in this section also show that the ambient density present in ground based tests has a measurable effect on the backflow currents caused by CEX ions. Simulations of a thruster operating in vacuum suggest that ground tests are likely to overpredict the backflow current which will actually occur under operational conditions.

Section 4: Conclusions and Future Work

A computational model of an SPT plasma plume has been constructing using a quasi-neutral PIC-DSMC model. This model is based on theoretical work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. The resulting axisymmetric simulation can model an SPT plume on meter length scales with run times of only 2-4 hours. Comparisons between the computational model and existing data highlight weaknesses in both the model and the data. The simulated and experimental results agree well when the extra currents shown in Table 3 are included in the computational model. These extra currents appear to represent both a lack of Xe^{++} in the simulation and the presence of external current sources in the experimental tests. Although further work is needed, the results suggest that the simulation is accurately modeling the physics of the plume region and that it can give predictions of backflow currents under operational conditions.

Future work is planned to improve this model and expand it to more realistic geometries. A model for Xe^{++} CEX collisions will be added in the near future and an effort will be made to evaluate the influence of the magnetic field in the region very close to the thruster exit. Surface interaction models will also be added to the simulation to allow prediction of ion impingement energies and erosion effects. In addition, we note that the present axisymmetric simulation does not allow modeling of realistic spacecraft geometries. We will address this issue through future work on a 3-D PIC-DSMC simulation. Such a simulation would allow direct modeling of plume interactions with meter scale spacecraft on workstation class machines. When mature, the PIC-DSMC model should be a useful tool for investigating the impact of SPT thrusters on commercial satellites.

Acknowledgments

The authors would like to thank Tom Randolph at Space Systems Loral and David Manzella at NYMS, Inc. for providing us with the data used in this paper. This paper was funded by the Air Force Office of Scientific Research.

References

[1] Absalamov, S.K., Andreev, V.B. et al. "Measurement of Plasma Parameters in the Stationary Plasma Thruster (SPT-100) Plume and Its Effect on Spacecraft Components," AIAA Paper 92-3156, July 1992.

[2] Manzella, D.H. "Stationary Plasma Thruster Plume Emissions," IEPC Paper 93-097, September 1993.

[3] Manzella, D.H. "Stationary Plasma Thruster Plume Velocity Distribution," AIAA Paper 94-3141, June 1994.

[4] Manzella, D.H. and Sankovic, J.M. "Hall Thruster Ion Beam Characterization," AIAA Paper 95-2927, July 1995.

[5] Myers, R. M. and Manzella, D.H. "Stationary Plasma Thruster Plume Characteristics," IEPC Paper 93-096, September 1993.

[6] Rhee, M.S. and Lewis, M.J. "Numerical Simulation of Stationary Plasma Thruster Exhaust Plume," AIAA Paper 95-2928, July 1995.

[7] Randolph, T. Space Systems Loral. Internal Memorandum.

[8] Chapman, B. *Glow Discharge Processes,* Wiley, New York, 1980, pg. 45.

[9] Banks, P. "Collision Frequencies and Energy Transfer, Electrons," *Planet. Space Sci.,* Vol. 14, 1966, pp. 1085-1101.

[10] Mitchner, M. and Kruger, C. *Partially Ionized Gases,* Wiley, 1973, pg. 58.

[11] Rapp, D. and Francis, W.E. "Charge Exchange Between Gaseous Ions and Atoms," *Journal of Chemical Physics*, Vol. 37, No. 11, 1962, pp. 2631-2645.

[12] Birdsall, C.K. and Langdon, A.B. *Plasma Physics Via Computer Simulation,* Adam Hilger, Bristol, 1991.

[13] Bird, G.A. *Molecular Gas Dynamics,* 1st ed., Clarendon Press, Oxford, 1976.

[14] Ruytan, W.M. "Density-Conserving Space Factors for Particle Simulations in Cylindrical and Spherical Coordinates," *J. Comp. Phys.,* Vol. 105, 1993, pp. 224-232.

[15] Gavryushin, V.M. and Kim, V. "Effect of the Characteristics of a Magnetic Field on the Parameters of an Ion Current at the Output of an Accelerator with Closed Electron Drift," *Sov. Phys. Tech. Phys.*, Vol. 26, No. 4, April 1981, pp. 505-507.

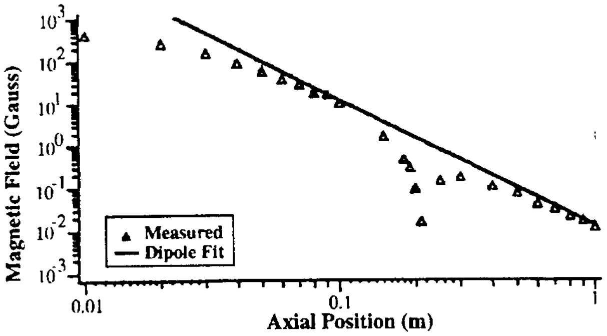


Figure 1: Magnetic Field Strength vs. Axial Position

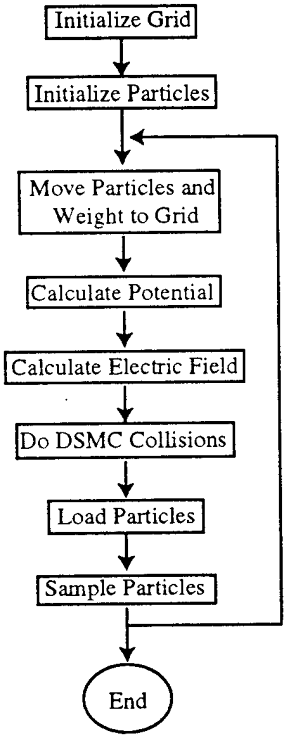


Figure 2: PIC-DSMC Simulation Algorithm

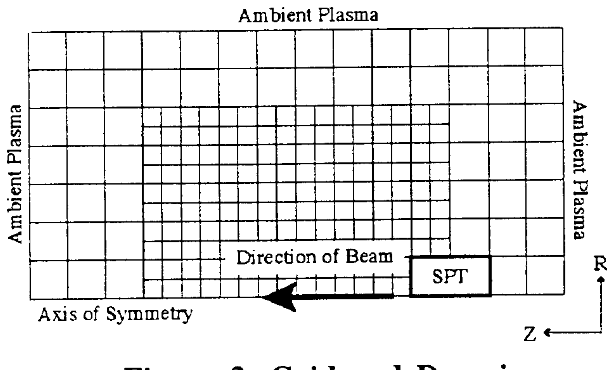


Figure 3: Grid and Domain

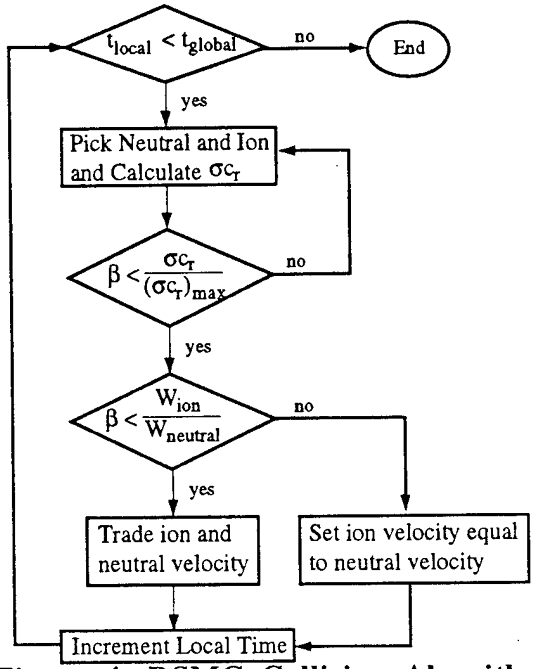

$t_{local} < t_{global}$
no
End
yes
Pick Neutral and Ion and Calculate σc_T
$\beta < \frac{\sigma c_T}{(\sigma c_T)_{max}}$
no
yes
$\beta < \frac{W_{ion}}{W_{neutral}}$
no
yes
Trade ion and neutral velocity
Set ion velocity equal to neutral velocity
Increment Local Time

Figure 4: DSMC Collision Algorithm

(Contours are 4 V apart)

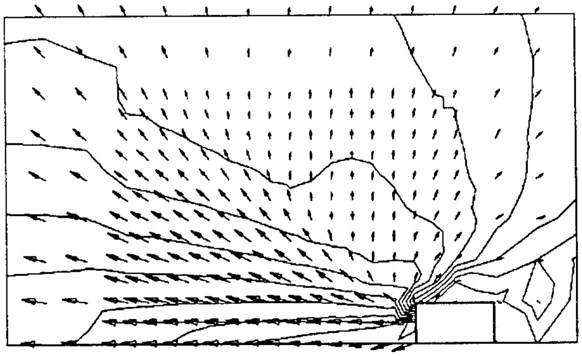


Figure 5: Contour Plot of Potential and Vector Plot of Current Density

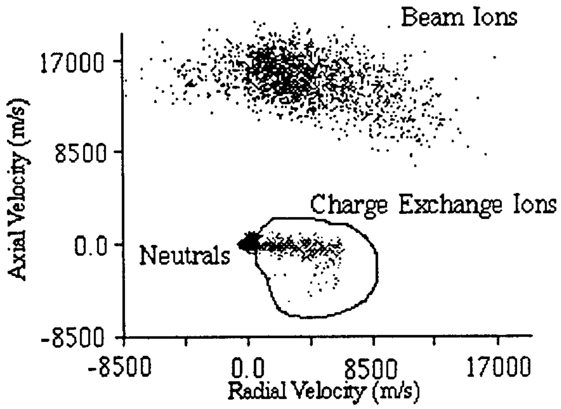


Figure 6: Phase Space Plot of Plume Ions

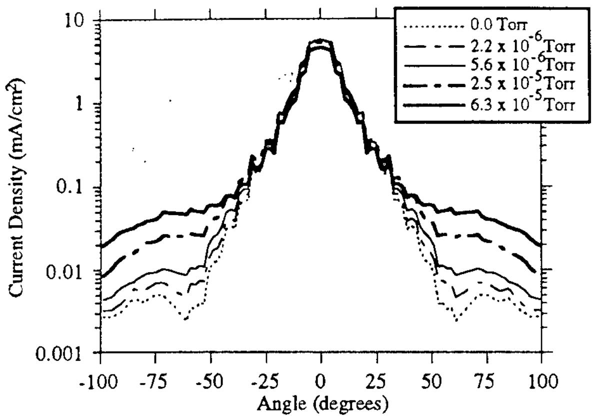
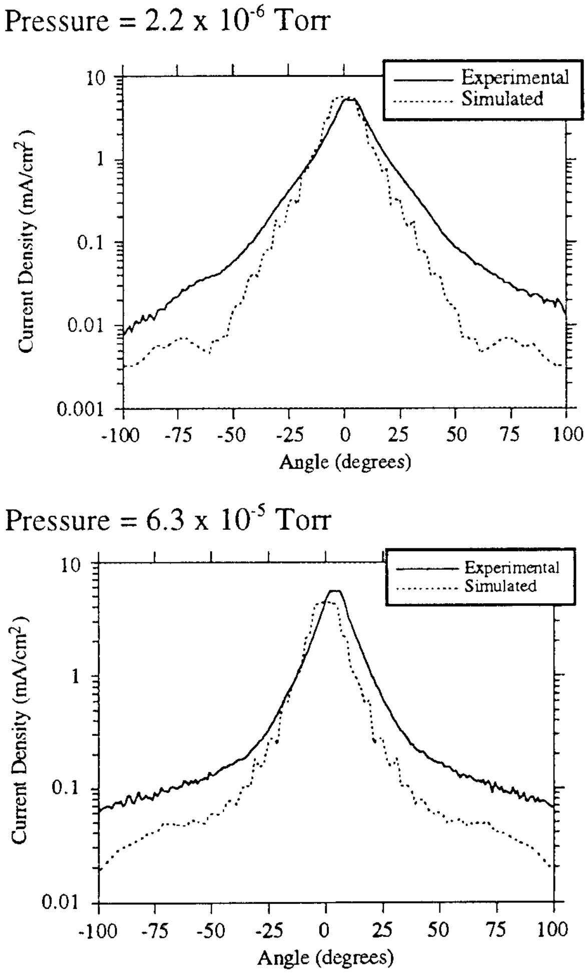


Figure 7: Simulated Ion Current Density as a Function of Ambient Pressure (z = 60 cm)



Pressure = 2.2 x 10^{-6} Torr

Pressure = 6.3 x 10^{-5} Torr

Figure 8: Comparison of Experimental and Simulated Ion Current Densities (z = 60 cm)

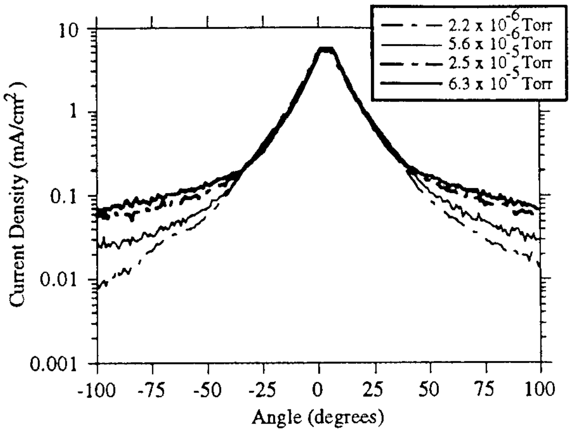
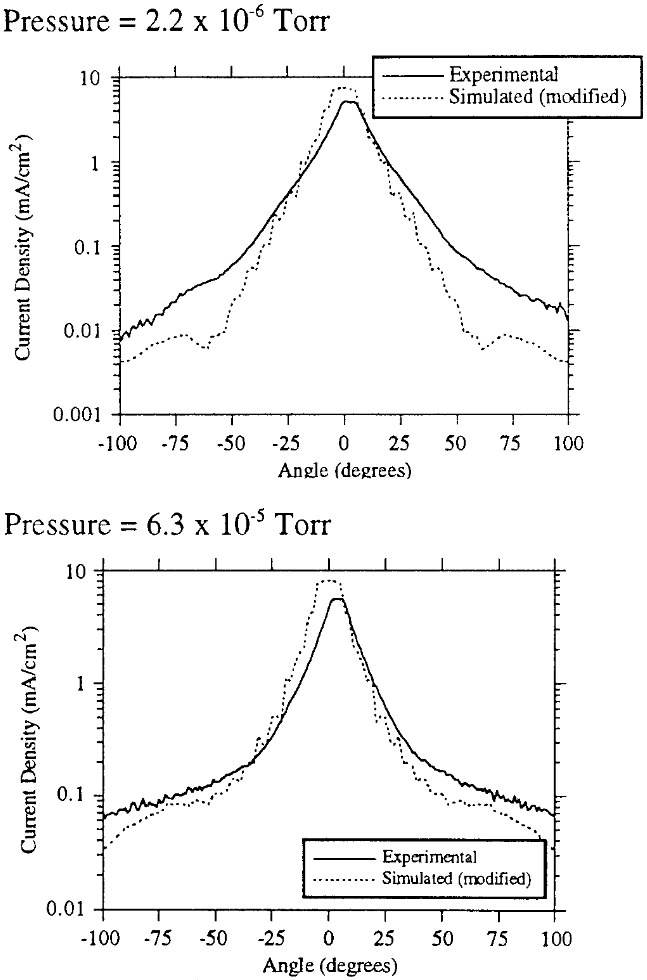


Figure 9: Measurements of Ion Current Density as a Function of Facility Pressure (z = 60 cm)[4]



Pressure = 2.2 x 10^{-6} Torr

Pressure = 6.3 x 10^{-5} Torr

Figure 10: Comparison of Experimental and Simulated Ion Current Densities with Modified Beam Currents (z = 60 cm)

AIAA 96-3299

Three Dimensional PIC-DSMC Simulations of Hall Thruster Plumes and Analysis for Realistic Spacecraft Configurations

D. Oh and D. Hastings
MIT Space Systems Laboratory
Cambridge, MA

32nd AIAA/ASME/SAE/ASEE
Joint Propulsion Conference
July 1-3, 1996 / Lake Buena Vista, FL

AIAA-96-3299

THREE DIMENSIONAL PIC-DSMC SIMULATIONS OF HALL THRUSTER PLUMES AND ANALYSIS FOR REALISTIC SPACECRAFT CONFIGURATIONS

David Y. Oh* and Daniel E. Hastings†
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Abstract

A computational model of a Stationary Plasma Thruster (SPT) has been developed using a quasi-neutral Particle-in-Cell/Direct Simulation Monte Carlo (PIC-DSMC) model. This model is based on previous work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. Axisymmetric and three dimensional versions of the model have been developed and validated against each other. The axisymmetric model is used to simulate a SPT-100 operating in vacuum, the neutral background present in ground based tests is shown to have a significant influence on the CEX collision rate. The neutral background is shown to cause experiments to overpredict the backflow from an SPT plume. Predictions are made for the sputter erosion rates of Silver, Quartz, and Silicon surfaces placed in the plume region. Results are presented to aid designers making "back of the envelope" calculations of surface erosion rates, and it is shown that the erosion rate scales with the square of the distance from the anode exit.

A three dimensional PIC-DSMC model has been used to simulate an SPT-100 thruster mounted on a typical communications satellite. A surface sputtering model is used to predict the impact the plume has on surfaces of the satellite. Results are presented for an SPT-100 mounted at different angles with respect to the solar array. The results show that the thruster's cant angle has a large impact on the rate of erosion and the area over which erosion occurs. Future work is planned to convert the PIC-DSMC model to support simulations on unstructured meshes.

Nomenclature

T_e	= Electron Temperature
$\bar{c}_e$	= Mean Electron Thermal Velocity
c_r	= Relative Speed between Collision Partners
e	= Elementary Charge = 1.6×10^{-19} C
k	= Boltzmann's Constant = 1.38×10^{-23} J/K
n_e	= Electron Number Density
n_o	= Reference Charge Density (arbitrary)
n_{ref}	= Reference Electron Number Density
z	= Axial Distance from Anode Exit
Γ_i	= Ion Flux
φ	= Electrical Potential

Introduction

When compared to chemical propulsion systems, electric propulsion offers substantial fuel savings for stationkeeping applications. One system which has shown particular promise is the Stationary Plasma Thruster (SPT), also referred to as the Hall Thruster. These devices have a near optimum specific impulse for north-south stationkeeping and are being actively marketed for use on Western satellites. However, although SPT's have an extensive Russia flight heritage, they have yet to fly on a Western satellite. Western designers have expressed concern that the plasma plume emitted by the SPT may erode and contaminate the surfaces of satellites and interfere with communications signals. These concerns must be addressed before SPT's can be used for commercial applications.

In order to study plume contamination issues, a substantial amount of experimental work has been conducted. Ion fluxes and distributions have been determined and plume induced sputtering and contamination have been measured in several ground experiments.[1-4] But at the same time, relatively little effort has been made to model processes occurring in the

* Research Assistant, Student Member, AIAA
†Professor of Aeronautic and Astronautics, Associate Fellow, AIAA

SPT's plume region. Those models which do exist are relatively simple and are not well suited to modeling complex satellite geometries.[5,6] More detailed models are needed to fully characterize the plume and to understand the relationship between existing experimental data and actual operating conditions.

This paper presents an advanced computational model of an expanding SPT plume. The model uses a combination of the Particle-in-Cell (PIC) and Direct Simulation Monte Carlo (DSMC) methods to model collisions in the plume and track the flow of ions and neutrals across the domain. The basic PIC-DSMC algorithm has been described in previous work.[7] Many improvements have been made to the model, including the use of double ions, the addition of surface sputtering, and the development of an improved source model. Axisymmetric and three dimensional versions of the code have been developed and validated against experimental data and the simulation has demonstrated the ability to model the plume on meter length scales. The resulting model runs on a UNIX workstation and should be useful to designers interested in evaluating the impact of an SPT on realistic satellite configurations.

An axisymmetric plume model is described in detail in a companion paper which compares simulated results to a variety of experimental data.[8] Simulated ion current density measurements are shown to match experimental data to within a factor of two to three on meter length scales. A surface erosion model is also presented and shown to agree well with experimental data, though the results depend heavily on the sputtering coefficient. This paper presents axisymmetric and three dimensional results which simulate conditions beyond those which can be studied in ground based experiments. These results include an axisymmetric simulation of an SPT-100 operating in vacuum and three dimensional simulations of a SPT thruster mounted on a communications satellite. The paper concentrates on modeling the SPT-100 because it is the most extensively studied of the Hall thrusters at this time. In principle, the model can be modified and applied to new Hall and anode layer (TAL) thrusters as data becomes available.

Section 2 summarizes the PIC-DSMC algorithm and describes aspects of the model which are unique to the three dimensional geometry, section 3 presents simulated results of an SPT-100 operating in vacuum, and section 4 presents results from a fully three dimensional simulation of an SPT-100 mounted on a communications satellite. Finally, section 5 presents conclusions and plans for future work.

Theory and Computational Method

The PIC-DSMC model used in this paper has been described in detail in previous work.[7] This section briefly summarizes the basic algorithm and describes aspects of the model specific to a three dimensional geometry. It focuses in particular on the boundary conditions used to simulate the surfaces of satellites and their interaction with the surface erosion model. Details of the underlying method, including the axisymmetric model, are given in a companion paper.[8]

The SPT-100 has been studied extensively in ground tests and many of its basic characteristics are well documented. Table 1 gives the parameters used to model the thruster in the PIC-DSMC plume model.

Inner Anode Diameter[1]	56 mm
Outer Anode Diameter[1]	100 mm
Cathode Orifice Diameter[2]	0.5 mm
Propellant	Xe
Propellant Flow Rate[2]	5.2 mg./sec.
Fraction of Propellant Directed to Cathode[2]	~10%
Electron Temperature[3] (Te)	2 eV
Axial Ion Velocity[1]	~17000 m/s
Fraction of Ions which are Double Ions[1]	~20%
Fraction of Propellant Ionized in Discharge Chamber[2]	> 95%

Table 1: SPT-100 Basic Characteristics

Xe propellant is ionized in the anode and emerges from the thruster in the form of neutrals, ions, and double ions. The following statements can be made about the plume region.[7]

- The Debye length is small, so the plume is quasi-neutral except near surfaces.
- The ions are unmagnetized.
- The electrons are unmagnetized when z > 25 cm.
- The electrons are collisionless.

Based on these observations, we have constructed a quasi-neutral PIC-DSMC simulation of a Hall Thruster plume.

The PIC-DSMC method uses macroparticles to statistically model gases at a molecular level. The particle equations of motion are integrated using the leapfrog method and the electric field is obtained from the electric potential, which is in turn obtained by inverting the Boltzmann relationship.

$$n_e = n_{ref} \exp(e\phi / kT_e)$$

This procedure is valid for isothermal, collisionless and unmagnetized electrons moving at low drift velocities. A constant electron temperature of 2 eV is assumed in these simulations.

Collisional processes are modeled between move steps using the DSMC method. Table 3 shows the collision processes included in the present PIC-DSMC model.

CEX	**Elastic**
Xe-Xe$^+$ Xe-Xe^{++}	Xe-Xe Xe-Xe$^+$ Xe-Xe^{++}

Table 3: Collisions covered in the simulation

Collisions between charged particles are not included because it is computationally impractical to simulate them at the present time.[8]

Particles are loaded into the simulation at each time step to simulate the flow from a Hall thruster. The ion and neutral distribution functions are determined from an empirical model of an SPT-100 which has been described in previous work.[8] This model is based on experimental data and contains no free parameters. Approximately 10% of the propellant entering an SPT-100 is diverted to the cathode. This flow is assumed to consist entirely of neutrals and is treated differently in axisymmetric and three dimensional domains. In a three dimensional geometry, the flow emerges from a cathode orifice which is placed 7.5 cm above and 1.0 cm downstream of the center of the anode exit. In an axisymmetric geometry, the cathode orifice can not be directly simulated due to geometric restrictions. The flow from the cathode is therefore diverted to the anode where it emerges as unionized propellant. Both cathode and anode neutrals are assumed to have a temperature of 1000 K and are choked at the cathode and anode exits respectively.

The major motivation for developing a three dimensional plume model is to model realistic spacecraft geometries. Figure 1 shows a three dimensional model of a commercial communications satellite. The three dimensional PIC-DSMC model is built on an embedded Cartesian mesh. A simulated bus, yoke, and solar array are shown on a 3.2 m x 4.4 m x 3.2 m computational domain, and an embedded grid is visible along the edge of the main bus. This grid is collocated with the thruster and is used to better resolve the core of the plume. The section of bus shown has dimensions of 1.1 m x 1 m x 2.6 m and represents a quarter of the spacecraft's main bus. A 1.9 meter yoke connects the bus to the end of a solar array. The array is 1.5 meters wide and continues off the top of the domain. Only the bottom 2.7 meters of the array is included in the simulation, and it is this area which should experience the most plume degradation. There are several limitations on the types of satellites which can be modeled with a 3D embedded grid. First, in order to improve computational efficiency, all surfaces in the domain must be grid conforming. This means that the satellite must be constructed from boxes that follow cell boundaries and precludes the use of curved surfaces to model antennae. Second, the weighting functions used along the boundaries between adjacent objects are calculated as though only one object were present. As a result, both charge and neutral densities are calculated incorrectly along the line which intersects both objects. However, the accuracy of the simulation away from this boundary is not affected. Third, the resolution of embedded grids is fundamentally limited by the underlying particle simulation. In conventional computational fluid dynamics, embedded meshes are used to resolve areas of interest. The location of the mesh is largely arbitrary and determined entirely by geometry. In particle simulations, however, a minimum number of particles should be present in each cell in the domain. Adding an embedded mesh without increasing the number of particles results in fewer particles per cell and actually reduces the accuracy of the simulation. Therefore, particles must be added along with the mesh to maintain the same level of accuracy. Embedded grids can be used in areas of high density, such as the core of the plume, but can not be used to provide better resolution of objects in the far field. The number of particles which can be added is restricted by available memory and prevents the arbitrary use of embedded grids to resolve objects of interest.

One issue of particular interest to satellite designers is the interaction of the plume with surfaces of a spacecraft. In order to study surface interaction issues, a surface sputtering model has been developed and incorporated into the PIC-DSMC model. A relatively simple model is used in axisymmetric geometries and is described in a companion paper.[8] A similar model is used for three dimensional simulations and is described below.

Just as in axisymmetric geometries, deposition effects are ignored in three dimensional simulations. This is consistent with experimental results that report little or no deposition on witness plates, even when they are placed at high angles with respect to the centerline.[9] The assumption is valid when modeling thrusters with ceramic anodes, in which eroded material is likely to be benign and can be safely ignored. This assumption is probably invalid for TAL thrusters, however, which have metal anodes. Work with ion thrusters has shown that eroded material from metal grids has a tendency to plate on exposed surfaces, thus causing deposition damage.[10] The SPT's primary impact is to cause erosion due to sputtering damage. In three dimensional geometries, any particle crossing an object boundary is removed or reflected as appropriate for that species. Ions are neutralized and removed from the simulation while neutrals are reflected back into the domain in a manner consistent with an ideal specular surface. When creating an object, it is necessary to specify its potential with respect to the plasma. A solar panel would be represented by an object of fixed potential. Different points on the array would be assigned different potentials to mimic the distribution of cells across the array. A thermal blanket or other electrically isolated object would be represented by surfaces with floating potentials. These objects would be treated as dielectrics whose potential is determined by balancing the flux of ions and electrons to the surface.

When a plasma interacts with an object, a sheath typically forms near exposed surfaces. Figure 2 shows the structure of the sheath near a surface which is at the floating potential. This sheath is a non-quasi neutral region with a width on the order of the Debye length. The present PIC-DSMC model is quasi-neutral by assumption, and can not directly simulate this region. Fortunately, modeling the sheath itself is unnecessary. The Bohm sheath criterion requires that ions entering the sheath must be moving faster than the ion acoustic velocity. Once an ion enters the sheath, it is continuously accelerated until it strikes the wall. The sheath/pre-sheath boundary therefore represents a sink into which ions disappear and never reemerge. The quasi-neutral PIC algorithm can model the flow of ions in the pre-sheath region. Once an ion leaves the pre-sheath and enters the sheath, it never reemerges, so the ion is removed from the simulation. The situation is analogous to a choked flow through a rocket nozzle. Because no information passes from the sheath back into the pre-sheath region it is possible to simulate the pre-sheath without knowing the structure of the sheath itself.

For surfaces of fixed potential, the boundary conditions are more complex. When the surface potential is less than the local plasma potential, the pre-sheath can not "see" particles in the sheath region and the magnitude of the potential drop is irrelevant. When the surface potential is greater than the plasma potential, the electric field is reversed and the sheath will reflect some ions away from the wall. Again, the quasi-neutral PIC method can accurately simulate the pre-sheath region, but an analytic model must be applied to simulate the sheath itself. All ions striking a surface are assumed to cross into the sheath region, but some are reflected by the sheath and others are neutralized by the surface. The potential rise across the sheath is calculated by subtracting the local plasma potential from the surface potential. The energy of the incoming particle is then compared to the sheath potential. If the particle's kinetic energy is greater than the sheath potential, the ion is removed from the simulation. If the particle's energy is less than the sheath potential, the particle is treated like a neutral and reflected back into the domain. As a result, low energy ions are reflected while high energy ions are neutralized by the underlying surface. This accurately reflects the behavior of ions when they encounter a positively charged surface.

This surface model is only valid when conventional 1D sheath theory applies. This is generally the case when the Debye length is much less than the length scale of the surface, which is a restatement of the quasi-neutral assumption. The surface model is therefore invalid in wakes and other regions of low charge density. Errors may also be introduced along leading edges when the plasma flow is primarily parallel to the surface of interest.[11] The surface model also models electrons in the pre-sheath using the Boltzmann relationship. Although this assumption is typical of 1D sheath theory, it is not strictly accurate for electrons with non-zero drift velocities. Nevertheless, we represent electrons using the Boltzmann relationship.

Sputtering rates are calculated by tabulating the material removed by each ion and neutral which strikes a simulated surface. The amount of material lost is determined by multiplying an energy dependent sputtering coefficient by the macroparticle weighting factor. Multiple materials can be simulated by calculating separate sputtering coefficients, and no post-processing is necessary. A particle's impact energy is given by the sum of its kinetic energy and the energy it gained or lost in the sheath. All particles are assumed to enter normal to the surface and neutrals undergo no acceleration in the sheath region. The sheath drop is calculated differently on fixed and floating surfaces. On fixed surfaces, the drop is determined by subtracting the surface potential from the local plasma potential. On floating surfaces, the drop is determined from 1D sheath theory and is given by

$$\Phi_f = \frac{kT_e}{e}\ln\left(\frac{4\Gamma_i}{n_o\bar{c}_e}\right)$$

This paper presents simulated erosion rates for three different materials. The first material is Silver, which is commonly used for solar cell interconnectors. The following linear fit was used for its sputtering coefficient.[12]

$$S = 7.334\times10^{-3}E - 0.29511$$

The second material is Quartz, which is used to model solar cell coverglasses. Its sputtering coefficient is given by[9]

$$S = 7.105\times10^{-4}E - 0.01815$$

The sputtering coefficient for the anti-reflective coating present on the coverglass is unknown. The third material modeled was Silicon. Its sputtering coefficient is given by[12]

$$S = 1.0577\times10^{-3}E - 0.12115$$

In all cases, the sputtering coefficient is for Xenon ions striking normal to the surface. It has been observed that the atoms striking at different angles may have higher sputtering coefficients.[13] At the present time, however, the angular dependence for Xenon sputtering Silver, Quartz, and Silicon is unknown. As a result, this surface model may underpredict the actual erosion rate. Better experimental measurements of the sputtering coefficient are needed to improve the surface model.

In summary, axisymmetric and three dimensional PIC-DSMC simulations have been constructed based on the observation that the plume is a quasi-neutral, unmagnetized plasma in which the electrons are effectively collisionless. A variety of different collisional phenomena are included in the models, and an SPT-100 thruster is simulated using an empirical source model which is based on experimental data. A surface interaction model has also been developed for the 3D simulation. This model calculates the energy of particles striking the surface of a spacecraft and predicts the erosion rate for materials placed in the plume region. The final simulation was written in ANSI C, and simulations were conducted on Digital, IBM, and Hewlett-Packard workstations with 96-256 Megabytes of RAM. Run times were typically 2-4 hours for axisymmetric geometries and 12-15 hours for the three dimensional geometries.

Axisymmetric Results and Discussion

Simulations of axisymmetric geometries have been used to study the plume and to verify the underlying PIC-DSMC algorithm. A separate paper details efforts to validate the axisymmetric model by duplicating ground based experiments.[8] This section presents additional results which extend the model to conditions which can not be duplicated in the laboratory.

Ground based tests are generally carried out in vacuum chambers where the quality of the vacuum is limited by the facility's pumping speed. SPT tests are typically carried out at pressures of 3 x 10^{-5} Torr, though pressures as low as 6 x 10^{-6} Torr can be achieved in some facilities. Figure 3 shows several plots of the ion current density 60 cm from the exit of an SPT-100 thruster. One plot shows the experimentally measured ion current density at a pressure of 2.5 x 10^{-5} Torr.[3] The next set shows an axisymmetric simulation of a thruster operating at the same ambient pressure. The experimental and simulated results are within a factor of 2-3 across most of the domain. The third set of results is a simulation of a thruster operating in vacuum. At high angles, the ion current density in vacuum is almost an order of magnitude lower than in simulated ground tests. These results show that the ambient neutrals present in ground based tests can significantly effect experimental results. These neutrals increase the CEX collision rate and lead to experiments that overpredict the backflow which will occur on orbit. These simulations suggest that significantly less backflow will be present under operational conditions. This may allow designers to use and orient thrusters more aggressively than they would based on ground based tests.

The question of most interest to designers is how much damage the plume will cause to satellite surfaces. Figure 4 shows the predicted erosion rate for silver and quartz surfaces placed 60 cm from the exit of an SPT-100 operating in vacuum. The results are presented in units of meters per month (720 hours) of operation. Figure 4 shows that the estimated erosion rate varies very strongly with angle from the centerline and varies by roughly an order of magnitude for each 20 degree change in angle. This indicates that a thruster's angle with respect to a surface will have a huge impact on the erosion rate. Another question of interest is how the erosion rate scales with distance from the anode exit. The solid and dotted lines in figure 5 show simulated erosion rates as measured 1 m from the thruster exit. These rates are considerably lower than those shown in figure 4, though silver still erodes 10 microns/month at angles as high as 55 degrees from the centerline. Superimposed on the simulated rates 1 m from the exit are the rates from Figure 4 scaled by $1/z^2$. The results show very good agreement, and suggest that it should be possible to scale observed erosion rates using a $1/z^2$ law. The results shown in Figure 4 can be combined with this scaling law to produce quick "back of the envelope" estimates of the erosion rate at a given position relative to the thruster. Designers can also use the relationship to scale experimentally measured erosion rates to areas far away from the anode exit.

3D Results and Discussion

After the PIC-DSMC algorithm was validated experimentally, a three dimensional PIC-DSMC plume model was developed and validated against the axisymmetric results. Figure 6 shows several plots of the ion current density 60 cm from the thruster exit at a pressure of 2.2×10^{-6} Torr. Three types of results are overlaid on the graph: an axisymmetric simulation (marked as "2D"), a three dimensional simulation, and experimental data from Manzella.[3] The simulated results agree well with each other and confirm that the change from an axisymmetric to a three dimensional geometry has not significantly affected the simulation. Both sets of results show good agreement with data, again matching to within a factor of 2-3 across the domain. Similar comparisons were made at different pressures to validate the 3D code, and all show the same level of agreement. One interesting aspect of the 3D results is that the number of particles per cell varies drastically from 1600 per cell in the core of the plume to less than 5 per cell at the edges of the domain. Traditionally, both PIC and DSMC schemes require 10 particles per cell to retain statistical accuracy. However, when the 3D results created with 3-1600 particles per cell are compared to axisymmetric results produced with 20-2000 particles per cell, the results are virtually identical. This implies that having less than 10 particles per cell at the edges of the plume does not hurt the accuracy of the simulation. This is the case because most of the important physics occurs in the center of the plume, where the majority of the CEX collisions actually occur. Particles at the edges of the plume undergo relatively few collisions and follow almost ballistic trajectories. Therefore, while it is important to maintain the number of particles per cell in the core of the plume, some latitude can be exercised in the outer parts of the 3D simulation.

Once the code was validated, an effort was made to study the effect the plume of an SPT-100 would have on a realistic satellite configuration. Simulations were conducted on a simple GEO comsat as shown in Figure 1. The configuration consists of a Bus, Yoke and Solar array which are located on a 3.2 m x 4.4 m x 3.2 m domain. The dimensions of the spacecraft are discussed in section 2 above. The spacecraft was assumed to be oriented with the arrays on the North and South sides of the spacecraft, so thrusters operating for N-S stationkeeping would fire in a vertical direction. The SPT-100 thruster was assumed to be mounted at the edge of the bus under the edge of the solar array as shown in Figure 1. The thruster's orientation was then varied during the simulations based on two parameters: a cant angle and an array angle. These angles are defined as shown in Figure 7. The cant angle is the angle between the thrust vector and the vertical axis. Lower cant angles better orient the thrusters for N-S stationkeeping, so higher cant angles represent an effective loss of I_{sp}. The array angle is defined as the solar panel's angle relative to an imaginary line connecting the yoke to the thruster. On a real spacecraft, the position of the thruster is fixed and the array turns with respect to the thruster. In our model, the array can only be turned in 90 degree increments. Therefore, to simulate different array angles, the thruster was mounted at different positions on top of the bus. This allows one to simulate a rotating array and still create a grid conforming body. All simulations were run for 15000 time steps with a time step of 0.1 normalized units. The potential of the entire surface of the array was fixed at -92 volts relative to the center of the plume. A real array would be covered by cells with potentials that vary from 0 to 92 Volts. Setting the entire surface to -92 V therefore represents a worst case in which all parts of the array are assumed to sit at a very negative potential with respect to the spacecraft.

A cant angle of 45 degrees and an array angle of 45 degrees were chosen as the spacecraft's baseline configuration. Figure 8 shows the baseline configuration overlaid by a surface of constant potential. The plume is clearly visible as a cone emerging from the top of the bus and oriented away from the solar array. Figure 9 shows a contour plot of ion current density on the face of the array nearest to the plume. Although the plot is noisy, a small but noticeable flux of ions is clearly reaching the surface even though the thruster is oriented away from the array. As one would expect, the area of highest flux is in the corner of the array which sits closest to the anode exit. Very little current reaches the lower or upper left corners of the array. Figure 10 is a contour plot showing the energy of ions striking the surface of the array. It shows that the ions with relatively high energies are actually striking the upper right corner of the array, while ions with relatively low energy strike the corner nearest to the thruster. This occurs because high energy ions follow relatively straight trajectories and do not turn far enough to strike the lower part of the array. CEX ions, on the other hand, have a small turning radius and are easily influenced by electric fields at the edge of the plume. These ions turn quite sharply and end up striking the bottom of the array.

Figures 11,12, and 13 show the calculated erosion rates for Silver, Silicon, and Quartz surfaces of the array. All materials are assumed to sit at -92 volts with respect to the center of the plume. These figures show that a noticeable and potentially significant amount of erosion will occur to interconnectors and coverglasses on the solar array. As one would expect, the highest erosion rates occur on surfaces closest to the thruster. Silver in particular has an erosion rate greater than 1 micron per month in some parts of the array. Quartz glass shows the next highest erosion rate, about 0.4 microns per month, and silicon is the most sputter resistant material, with a maximum rate of only 0.142 microns per month. The actual area over which these high rates occur is relatively small, covering an area of no more than 0.25 m^2. In addition, it should be noted that these surfaces are being held at a negative potential with respect to the plume. One obvious way to lower the peak erosion rate is to bias cells at the corners of the array positive with respect to the spacecraft. This would lower the energy of ions striking the surface of the array and help mitigate sputtering losses.

Canting the thruster away from the satellite's N-S axis lowers the thruster's effective I_{sp}, so it is desirable to use as small a cant angle as practical. To investigate the effects of changing the cant angle, simulations were run with cant angles from 0 to 45 degrees. Figure 14 shows the simulated erosion rate of silver on the same surface shown in Figure 11, but with the thruster canted at only 30 degrees. The 15 degree change of angle has a significant impact on the erosion rates on the array. Not only is the peak sputtering rate more than 3 times higher than in Figure 11, but the area over which damage occurs extends much farther up the side of the array. A plume shield would almost certainly be required to limit damage to the array. Figure 15 shows the sputtering rate of silver on the same surface with a cant angle of zero degrees. The erosion rate is now over an order of magnitude higher than in Figure 14, and is clearly unacceptable. The area of maximum damage now extends up the array to the end of the computational domain. The damage would certainly extend further if the domain continued to the end of the array.

In summary, the axisymmetric PIC-DSMC model has been successfully extended to three dimensions and validated though comparison to axisymmetric results. The model has been applied to a realistic satellite configuration, and a series of results have been shown which demonstrate the model's ability to evaluate thrusters placed in a variety of different orientations. The results shown were created in 12-15 hours on UNIX workstations, and the model offers designers the ability to simulate a variety of configurations in a relatively small amount of time. The accuracy of the underlying algorithm has also been verified through comparison to experimental data. These efforts are discussed in detail in a companion paper.[8]

Conclusions

A computational model of an SPT plasma plume has been constructed using a quasi-neutral PIC-DSMC model. This model is based on theoretical work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. The resulting simulation can accurately model an SPT-100 plume on meter length scales. Both axisymmetric and three dimensional models have been developed and validated against experimental data. Comparisons with experimental results are presented in a companion paper. The results shown in this paper simulate conditions which can not be studied in ground based experiments. These results include an axisymmetric simulation of an SPT-100 operating in vacuum and three dimensional simulations of a SPT thruster mounted on a communications satellite. The results show that ground based tests may significantly overpredict the backflow caused by CEX collisions. The presence of neutrals in the vacuum tank increases the CEX collision rate and creates artificially high ion current densities at high angles. In addition, predictions are made of the erosion rates which will be experienced by materials exposed to a thruster operating in vacuum, and a $1/z^2$ scaling law is suggested as a means of extending near field sputtering measurements to far field areas. Finally, a series of three dimensional simulations were conducted to demonstrate the model's ability to evaluate realistic spacecraft configurations. The results show that the thruster's cant angle has a strong impact on simulated erosion rates and the area over which erosion occurs. More importantly, the results demonstrate that the three dimensional model can be adapted and used by spacecraft designers to evaluate realistic spacecraft configurations. This will help designers to quickly evaluate the impact of Hall Thrusters on a given spacecraft design.

Future work is planned to improve the existing plume model. In particular, the incorporation of an embedded mesh into the PIC-DSMC simulation would allow designers to simulate irregular surfaces such as parabolic antennae. In addition, efforts will be made to improve the existing plasma source model and new sputter coefficients will be incorporated into the surface model as the data becomes available.

Acknowledgments

The authors which to thank David Manzella at NYMS Inc. for providing data used in this paper and Tom Randolph at SS/Loral for his help constructing a realistic satellite geometry. This paper was funded by the Air Force Office of Scientific Research.

References

[1] Manzella, D.H. "Stationary Plasma Thruster Plume Emissions," IEPC Paper 93-097, September 1993.

[2] Arkhipov, B.A. and Kozubsky, K.N. "The Development of the Cathodes-Compensators for Stationary Plasma Thrusters in USSR," IEPC Paper 91-023, October 1991.

[3] Myers, R.M. and Manzella, D.H. "Stationary Plasma Thruster Plume Characteristics," IEPC Paper 93-096, September 1993.

[4] Manzella, D.H. and Sankovic, J.M. "Hall Thruster Ion Beam Characterization," AIAA Paper 95-2927, July 1995.

[5] Rhee, M.S. and Lewis, M.J. "Numerical Simulation of Stationary Plasma Thruster Exhaust Plume," AIAA Paper 95-2928, July 1995.

[6] Dickens, J., Kirstiansen, M. and O'Hair, E. "Plume Model of Hall Effect Plasma Thrusters with Temporal Consideration," IEPC Paper 95-171, September 1995.

[7] Oh, D. and Hastings, D. "Axisymmetric PIC-DSMC Simulations of SPT Plumes," IEPC Paper 95-160, September 1995.

[8] Oh, D. and Hastings, D. "Experimental Verification of a PIC-DSMC Model for Hall Thruster Plumes," AIAA Paper 96-3196, June 1996.

[9] Randolph, T., Pencil, E. and Manzella, D. "Far-Field Plume Contamination and Sputtering of the Stationary Plasma Thruster," AIAA Paper 94-2855.

[10] Samanta Roy, R.I. "Numerical Simulation of Ion Thruster Plume Backflow for Spacecraft Contamination Assessment," Ph.D. Thesis, MIT Department of Aeronautics and Astronautics, June 1995.

[11] Wang, Joe. "Electrodynamic Interactions Between Charged Space Systems and the Ionospheric Plasma Environment," Ph.D. Thesis, MIT Department of Aeronautics and Astronautics, June 1991.

[12] Rosenberg, D. and Wehner, G.K. "Sputtering Yields for Low Energy He+-Kr+-, and Xe+-Ion Bombardment," Journal of Applied Physics, Vol. 33, No. 5, May 1962, pp. 1842-1845.

[13] Chapman, Brian. *Glow Discharge Processes: Sputtering and Plasma Etching*. Wiley and Sons, New York, 1980. pg. 247.

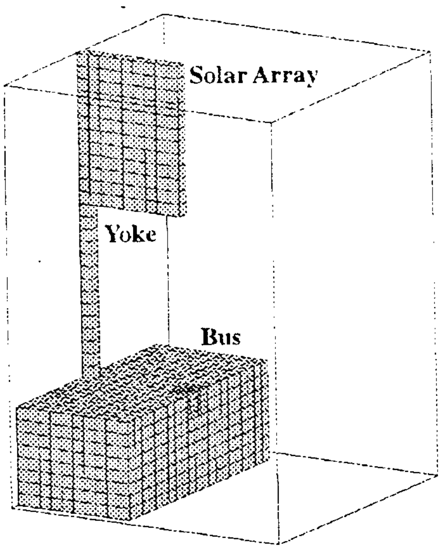


Figure 1: Model of a Communications Satellite on a 3D Embedded Mesh

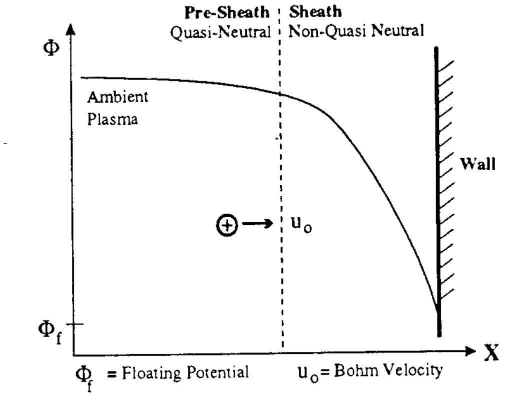


Figure 2: Plasma Potential Profile Near a Solid Surface at the Floating Potential

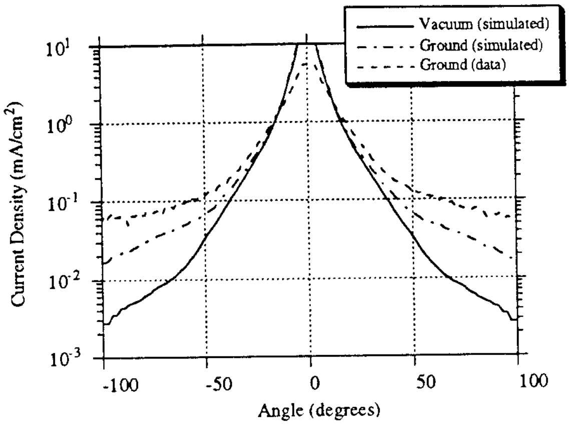


Figure 3: Ion Current Density 60 cm from Anode Exit on Ground (Pressure = 2.5 x 10^{-5} Torr) and in Vacuum

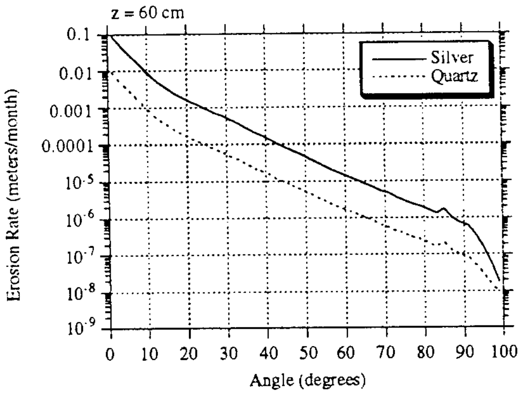


Figure 4: Predicted Erosion Rates for Silver and Quartz (z = 60 cm, Vacuum)

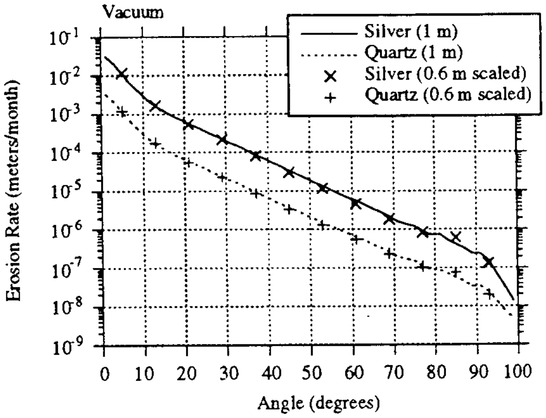


Figure 5: Comparison of Measured and Scaled Erosion Rates (z = 1 m, Vacuum)

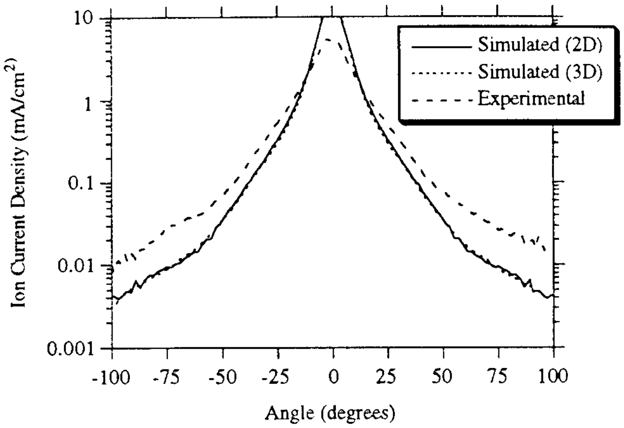


Figure 6: Comparison of Experimental, 2D, and 3D Simulations of Ion Current Density (z = 60 cm, P = 2.2 x 10^{-6} Torr)

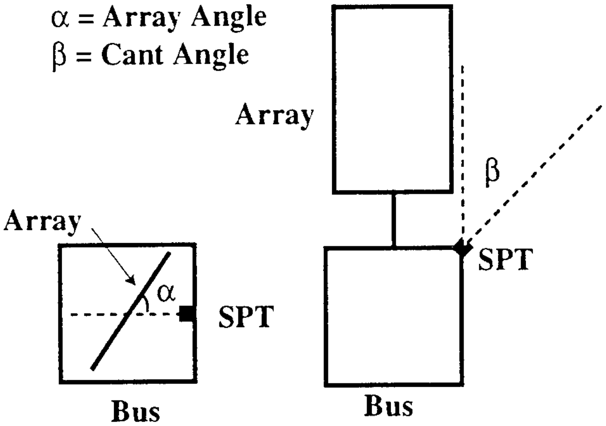


Figure 7: Definition of Cant Angle and Array Angle

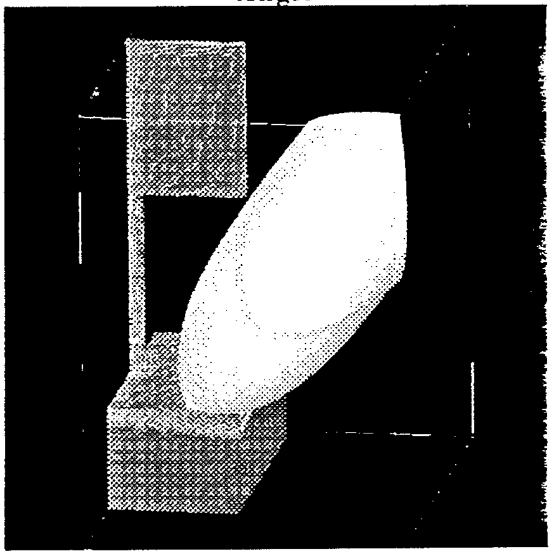


Figure 8: 3D Isopotential Surface Plot (Array Angle = 45°, Cant Angle = 45°)

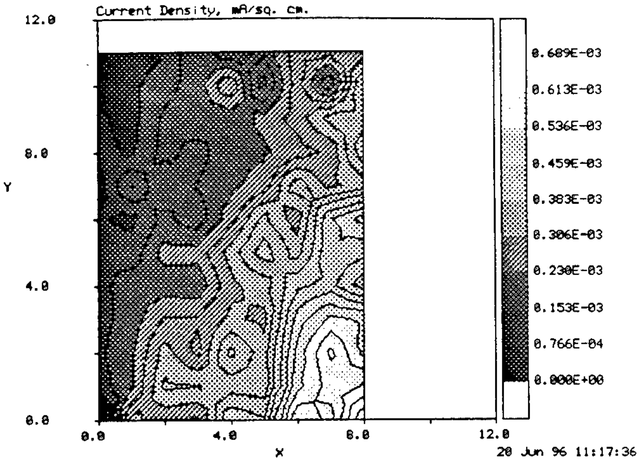


Figure 9: Ion Current Density on Surface of Solar Array (Array Angle = 45°, Cant Angle = 45°)

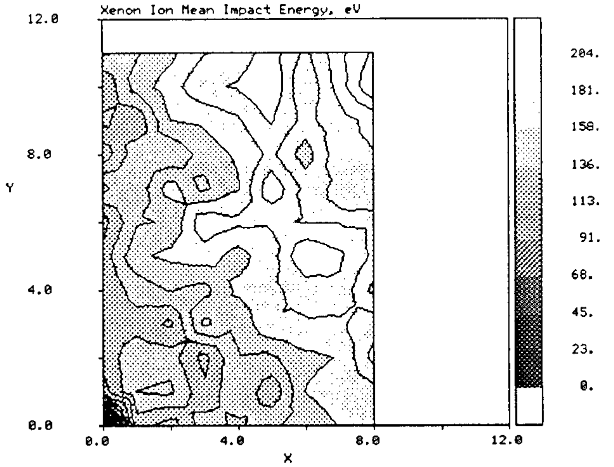


Figure 10: Mean Energy of Ions Striking Solar Array (Array Angle = 45°, Cant Angle = 45°)

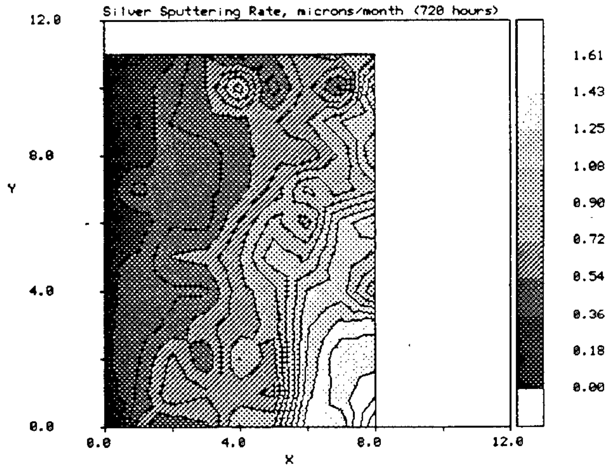


Figure 11: Simulated Erosion Rate for Silver (Array Angle = 45°, Cant Angle = 45°)

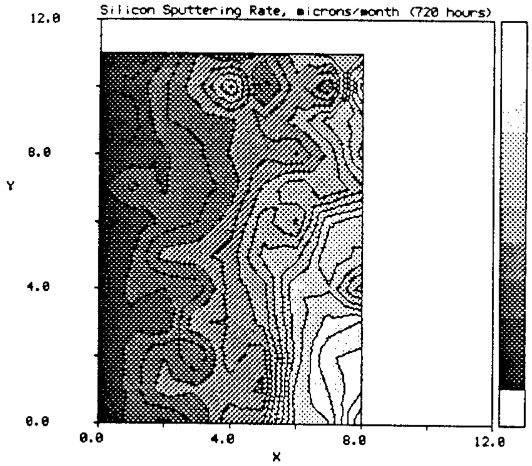

Silicon Sputtering Rate, microns/month (720 hours)

Figure 12: Simulated Erosion Rate for Silicon (Array Angle = 45°, Cant Angle = 45°)

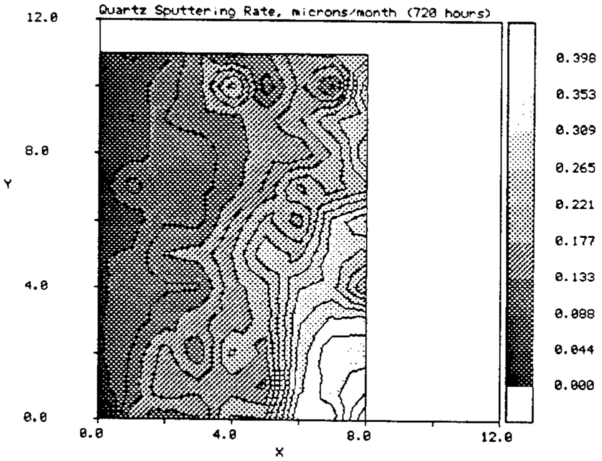
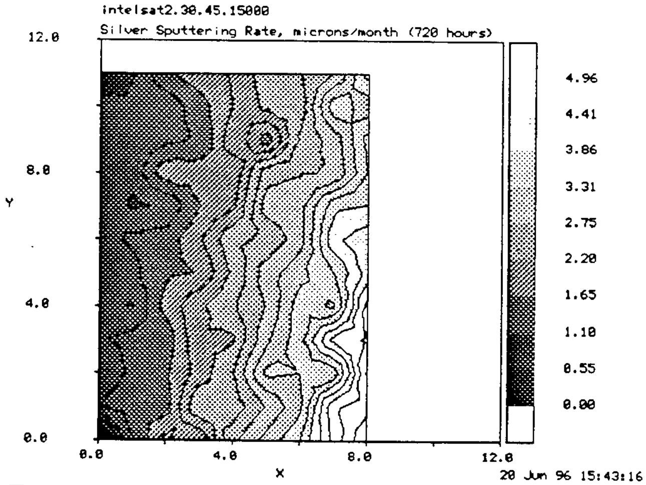
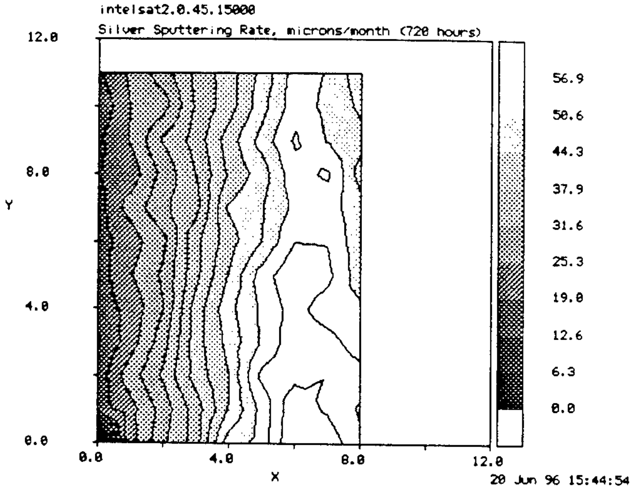

Quartz Sputtering Rate, microns/month (720 hours)

Figure 13: Simulated Erosion Rate for Quartz (Array Angle = 45°, Cant Angle = 45°)


intelsat2.30.45.15000
Silver Sputtering Rate, microns/month (720 hours)

20 Jun 96 15:43:16

Figure 14: Simulated Erosion Rate for Silver (Array angle = 45°, Cant angle = 30°)


intelsat2.0.45.15000
Silver Sputtering Rate, microns/month (720 hours)

20 Jun 96 15:44:54

Figure 15: Simulated Erosion Rate for Silver (Array angle = 45°, Cant angle = 0°)

AIAA 96-3196
Experimental Verification of a PIC-DSMC Model for Hall Thruster Plumes

D. Oh and D. Hastings
MIT Space Systems Laboratory
Cambridge, MA

32nd AIAA/ASME/SAE/ASEE
Joint Propulsion Conference
July 1-3, 1996 / Lake Buena Vista, FL

AIAA-96-3196

EXPERIMENTAL VERIFICATION OF A PIC-DSMC MODEL FOR HALL THRUSTER PLUMES

David Y. Oh* and Daniel E. Hastings†
Department of Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA

Abstract

A computational model of a Stationary Plasma Thruster (SPT) has been developed using a quasi-neutral Particle-in-Cell/Direct Simulation Monte Carlo (PIC-DSMC) model. This model is based on theoretical work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. Axisymmetric and three dimensional versions of the model have been developed and validated against a range of experimental data. Results from the axisymmetric simulation are compared to ion current density, surface erosion rate, and retarding potential analyzer data taken from the SPT-100 literature. The results show that the PIC-DSMC model compares well with ion current density and erosion rate data, to within a factor of 2-3 across most of the domain. Based on the axisymmetric results, an explanation is given for the formation of potential wings in ion thruster plumes and their absence in SPT plumes. The model is also shown to compare less well with RPA data. The reasons for the disagreement are unclear, and some inconsistencies seem to the present in the data as well as the simulation. An evaluation of the RPA data suggests that instabilities may play an important role in the formation of high energy ions. Future work is planned to convert the PIC-DSMC model so it supports simulations on unstructured meshes.

Nomenclature

T_e = Electron Temperature
$\bar{c}_e$ = Mean Electron Thermal Velocity
c_r = Relative Speed between Collision Partners
e = Elementary Charge = 1.6×10^{-19} C
k = Boltzman's Constant = 1.38×10^{-23} J/K
n_e = Electron Number Density
n_o = Reference Charge Density (arbitrary)
n_{ref} = Reference Electron Number Density
z = Axial Distance from Anode Exit
Γ_i = Ion Flux
λ_d = Debye Length
λ_{en} = Electron-Neutral Mean Free Path
λ_{ee} = Electron-Electron Mean Free Path
λ_{ei} = Electron-Ion Mean Free Path
φ = Electrical Potential
ρ_{ge} = Electron Gyro Radius
ρ_{gi} = Ion Gyro Radius
σ_{CEX} = Charge Exchange Collision Cross Section
σ_E = Elastic Collision Cross Section
χ = Particle Scattering Angle

Introduction

When compared to chemical propulsion systems, electric propulsion offers substantial fuel savings for stationkeeping applications. One system which has shown particular promise is the Stationary Plasma Thruster (SPT), also referred to as the Hall Thruster. This device has a near optimum specific impulse for north-south stationkeeping and is currently being marketed for use on Western satellites. However, although SPT's have an extensive Russian flight heritage, they have yet to fly on a Western satellite. Western designers have expressed concern that the plasma plume emitted by the SPT may erode and contaminate satellite surfaces and interfere with communications signals. These concerns must be addressed before SPT's can be used on commercial satellites.

In order to study plume contamination issues, a substantial amount of experimental work has been conducted. Ion fluxes and distributions have been

* Research Assistant, Student Member AIAA
†Professor of Aeronautics and Astronautics, Associate Fellow, AIAA

determined and plume induced sputtering and contamination have been measured in several ground experiments.[5,7,8,16,19] But at the same time, relatively little effort has been made to model processes occurring in the SPT's plume region. Those models which do exist are relatively simple and are not well suited for modeling complex satellite geometries.[1,2] More detailed models are needed to fully characterize the plume region and to understand the relationship between existing experimental data and actual operating conditions.

This paper presents an advanced computational model of an expanding SPT plume. The model uses a combination of the Particle-in-Cell (PIC) and Direct Simulation Monte Carlo (DSMC) methods to model collisions in the plume region and track the flow of ions and neutrals across the domain. The basic PIC-DSMC algorithm has been described in previous work.[3] Many improvements have been made to the model, including the use of double ions, the addition of surface sputtering, and the development of an improved ion source model. Axisymmetric and three dimensional versions of the code have been developed and validated against existing experimental data and the simulation has demonstrated the ability to model the plume on meter length scales. The resulting model runs on a workstation and should be useful to designers interested in evaluating the impact of an SPT on realistic satellite configurations.

This paper describes the improved PIC-DSMC model in detail and compares the axisymmetric simulation to a variety of experimental data. Comparisons are made to ion current, ion energy, and plate erosion data, and simulated results are shown to compare favorably in most cases. A companion paper presents results which can not be duplicated in ground based tests.[4] This includes an axisymmetric simulation of an SPT-100 operating in vacuum and three dimensional simulations of an SPT thruster mounted on a communications satellite. The work to date concentrates on modeling the SPT-100 because this is the most extensively studied Hall thruster at this time. In principle, the model can be modified and applied to new Hall and anode layer (TAL) thrusters as experimental data becomes available.

Section 2 presents the theoretical basis for the computational model and describes the PIC-DSMC algorithm in detail. Section 3 presents axisymmetric results and compares them to existing experimental data. Section 4 presents conclusions and plans for future work.

Theory and Computational Method

The SPT-100 thruster has been studied extensively in ground tests and many of its basic characteristics are well documented. Table 1 gives the parameters used to model the thruster in the PIC-DSMC plume model.

Inner Anode Diameter[5]	56 mm
Outer Anode Diameter[5]	100 mm
Cathode Orifice Diameter[6]	0.5 mm
Propellant	Xe
Propellant Flow Rate[6]	5.2 mg./sec.
Fraction of Propellant Directed to Cathode[6]	~10%
Electron Temperature[7] (T_e)	2 eV
Axial Ion Velocity[5]	~17000 m/s
Fraction of Ions which are Double Ions[5]	~20%
Fraction of Propellant Ionized in Discharge Chamber [6]	> 95%

Table 1: SPT-100 Basic Characteristics

Xe propellant is ionized in the anode and emerges from the thruster in the form of neutrals, ions, and double ions. This creates a plasma with characteristics can be estimated based on a simple $1/r^2$ expansion model. The results of such an analysis are given in Table 2.[3]

	Radial Position	
	0.2 m	**1.0 m**
λ_d	0.03 cm	0.14 cm
ρ_{ge}	6.7 cm	670 cm
ρ_{gi}	230 m	>20000 m
λ_{en}	85 m	340 m
λ_{ee}	130 m	3200 m
λ_{ei}	270 m	6700 m

Table 2: Estimates of Fundamental Plasma Parameters in Plume

Based on Table 2, the following conclusions can be drawn.

- The Debye length is small, so the plume is quasi-neutral except near surfaces.

- The ions are unmagnetized.
- The electrons are unmagnetized when z > 25 cm.
- The electrons are collisionless.

Although the electrons are magnetized in a small region in front of the thruster, experimental work suggests that the overall plume structure may be insensitive to small changes in the magnetic field strength.[8] We therefore neglect the magnetic field throughout the plume region.

Based on these observations, we have constructed a PIC-DSMC simulation of a Hall Thruster plume. This algorithm has been described in detail in previous work.[3] The PIC-DSMC method uses macroparticles to statistically model gases at a molecular level. The particle equations of motion are integrated using the leapfrog method. Solving these equations requires knowledge of the local electric field, which in turn requires knowledge of the electric potential. In conventional PIC schemes, the potential is obtained by solving Poisson's equation. In a quasi-neutral plasma, however, the potential can be obtained by inverting the Boltzman relationship.

$$n_e = n_{ref} \exp\left(e\phi / kT_e\right) \qquad (1)$$

This procedure is valid for isothermal, collisionless and unmagnetized electrons moving at low drift velocities. A constant electron temperature of 2 eV is used in this model.

The potential can be calculated directly from the charge density based on equation (1). The charge density is obtained by weighting ion macroparticles to the nodes of an embedded grid. In an axisymmetric geometry, particles are weighted to the grid using the cylindrical weighting functions given by Ruytan.[9] In a three dimensional geometry, a first order volume weighting function is used instead. Because the model is quasi-neutral by assumption, the Debye length does not limit the size of grid cells in the domain This results in a method which can simulate meter scale areas with relatively little computational effort.

Collisional processes are modeled between move steps using the DSMC method. The variation used in this work is a multi-species local time counter method described by Elgin.[10] A selection-rejection scheme is used to choose collision pairs and a single local time counter is used to determine the collision frequency for all collision processes. Table 3 shows the collision processes included in the present PIC-DSMC model.

CEX	Elastic
Xe-Xe$^+$	Xe-Xe
Xe-Xe^{++}	Xe-Xe$^+$
	Xe-Xe^{++}

Table 3: Collisions covered in the simulation

Elastic collisions between charged particles are not included in the present work because collisions between charged particles are dominated by short range coulomb interactions. As a result, it is necessary to simulate many small collisions to determine the momentum transferred to each macroparticle. At the present time, it is computationally impractical to model this process. Collective long range coulomb interactions are included as part of the PIC algorithm.

The collision process which most effects the structure of the plume is change exchange (CEX). Ion-neutral charge exchange occurs when a "slow" neutral and a "fast" ion exchange an electron to create a "fast" neutral and a "slow" ion. These slow ions can be accelerated back toward the spacecraft, creating backflow and impingement problems. Although the CEX mean free path is quite large, the backflow from CEX can not be neglected. In order to simulate a given collision process, it is necessary to know its velocity dependent collision cross section. The Xe-Xe$^+$ CEX collision cross section was calculated by Rapp and Francis and is given by[11]

$$\sigma_{\mathrm{CEX}} = \left(\mathrm{k}_1 \ln \mathrm{c}_r + \mathrm{k}_2\right)^2 \cdot 10^{-20}\,\mathrm{m}^2$$

Where $k_1 = -0.8821$, $k_2 = 15.1262$, and c_r is in m/s. The Xe-Xe^{++} CEX cross section has been measured experimentally and is shown in figure 1.[12] A logarithmic fit to the measured cross section gives the following.

$$\sigma_{CEX} = \left(3.4069 \times 10^{-9} - 2.7038 \times 10^{-10} \ln c_r\right) m^2$$

Figure 1 also shows the cross section for CEX between single and double ions. This cross section is almost an order of magnitude smaller than the double ion-neutral CEX cross section and is therefore neglected. When a simulated CEX collision occurs, the ion and neutral particle velocity vectors are exchanged to create a "slow" ion and a "fast" neutral. If the simulated collision partners have different macroparticle weights, an additional Monte Carlo selection is made to determine whether the velocity of the particle with the higher weight should be modified. This ensures that the total energy and momentum are statistically conserved over many collisions.

Neutrals in the plume region will also undergo several different types of elastic collisions. In general, the mean free path for these processes is large with respect to features of interest. Nevertheless, where practical, we include elastic collisions in the simulation. This allows one to model hypothetical thrusters with high neutral flow rates. The variable soft sphere (VSS) model is used to model Xe-Xe neutral collisions in the SPT plume and gives the following collision cross section.[13]

$$\sigma_E = \left(2.117 \times 10^{-18} c_r^{-0.24}\right) m^2$$

When a collision occurs, the scattering angle is chosen using the following relationship.

$$\alpha = 1.107 c_r^{0.01944}$$

$$\chi = \cos^{-1}\left(2\beta^{1/\alpha} - 1\right) \qquad (2)$$

This is consistent with the VSS collision model. The new velocity vectors are determined using the equations for conservation of momentum and energy and equation (2).

The collision cross section for Xe-Xe$^+$ elastic collisions has been derived analytically and is given by[14]

$$\sigma_E = \left(8.28072 \times 10^{-16}/c_r\right) m^2$$

According to theory, the neutral-ion collision cross section is independent of charge state, so the same relationship gives the Xe-Xe^{++} elastic collision cross section.[15] When a neutral-ion collision occurs, the scattering angle is chosen randomly based on an isotropic scattering distribution. This is consistent with the dynamics of a "hard sphere" collision. As with CEX collisions, because collision partners can have different macroparticle weights, an additional Monte Carlo selection is made to determine whether the velocity of the particle with the higher weight should be modified in a given collision. This ensures that the total energy and total momentum are statistically conserved over many collisions.

It has proved impractical to simulate Xe-Xe elastic collisions when a neutral background is present in the simulation, so these collisions were not included when simulating ground based experiments. The difficulty comes from the need to mix particles with different macroparticle weights. When simulating a relatively dense neutral background, a larger macroparticle weight must be used for the neutrals than for ions to keep the background particles from overwhelming the ions present in the simulation. However, when a simulated collision occurs, the size of the local collision timestep is proportional to the macroparticle weight. Therefore, Xe-Xe collisions require collision timesteps much larger than that required for Xe-Ion collisions. In practice, each neutral-neutral collision results in a collision timestep equivalent to hundreds of simulation timesteps, which effectively removes that cell from the DSMC part of the simulation. Xe-Xe collisions are therefore only included in simulations without a neutral background, where the neutral macroparticle weight is approximately equal to the ion macroparticle weight.

Particles are loaded into the simulation at each time step to simulate the exit flow from a Hall thruster. The ion and neutral distributions are determined based on an empirical model of an SPT-100 which has been described in previous work.[3] This model is based on experimental data and contains no free parameters. Two modifications have been made to the previously described SPT-100 source model. First, the mean ion drift velocity in the r/z plane has been changed from 16000 m/s to 17020 m/s. This high value corresponds to the Isp of an SPT-100 after excluding the flow to the cathode.[8] Second, the thermal component of the axial velocity is now based on a temperature of 34 eV rather than 3.4 eV. The effects of this change are discussed in section 3 below.

Approximately 10% of the propellant entering an SPT-100 is diverted to the cathode. The simulation treats this flow differently in axisymmetric and three dimensional geometries. In both cases, the propellant flowing through the cathode is assumed to consist entirely of neutrals. In the three dimensional model, the flow emerges from an orifice which is placed 7.5 cm above and 1.0 cm downstream of the center of the anode exit. In the axisymmetric model, the cathode orifice can not be simulated due to geometric restrictions. The cathode flow was therefore assumed to flow through the anode as unionized propellant. Both cathode and anode neutrals are assumed to have a temperature of 1000 K and are choked at the cathode and anode exits.

One issue of particular interest to satellite designers is plume induced deposition and/or erosion on sensitive parts of the spacecraft. Previous work has shown that eroded metal from the acceleration grids of ion thrusters can lead to substantial deposition on solar cell cover glasses.[14] The anode of an SPT thruster also undergoes substantial erosion, but the material released is a ceramic Boron Nitride compound which is unlikely to react with exposed surfaces. Since the propellant is also a noble gas and is unlikely to deposit onto surfaces, we neglect plume induced deposition in our surface interaction model. This assumption is consistent with existing experimental results.[16] It should be noted that this assumption may not be valid for the TAL thruster, which has a metal anode. This anode may introduce relatively reactive material into the plume region.

The SPT plume will cause sputtering to occur on exposed surfaces. In the axisymmetric model, sputtering erosion rates are calculated by recording the flux and energy of macroparticles crossing an arc at a fixed distance from the anode exit. The particles themselves are not neutralized by the surface, but continue on as though the surface were not present. This causes the simulation to slightly underestimate the acceleration of ions in the presheath, but has little impact on the final result. A more detailed discussion of plasma-surface interaction models is included in a companion paper. The flux and energy information are then post-processed to determine the magnitude of the sheath drop which would be present on a surface sitting at the floating potential. The magnitude of this drop is given by

$$\Phi_f = \frac{kT_e}{e}\ln\left(\frac{4\Gamma_i}{n_o\bar{c}_e}\right)$$

The sheath drop is then added to the recorded macroparticle energy to give the impact energy of ions on the surface. Once the impact energy is known, the amount of material removed can be determined from an experimentally measured energy dependent sputtering coefficient. This coefficient is material dependent and must be determined by experiment. Since Xe is the dominant species in the plume region, only Xe induced sputtering is considered in the model. One material of interest to designers is silver, which is used to make solar cell interconnectors. Its sputtering coefficient is given by[17]

$$S = 7.334\times10^{-3}E - 0.29511$$

Another material of interest is the glass used to cover solar cells. Solar cell coverglasses are covered by a thin anti-reflective coating whose sputtering coefficient is unknown. The exact composition of the coverglass varies, but the sputtering coefficient for Argon on quartz glass was used to represent a generic glass surface. It is given by.[16]

$$S = 7.105\times10^{-4}E - 0.01815$$

A more advanced surface model was used for three dimensional geometries and is described in a companion paper.[4]

In summary, a PIC-DSMC simulation has been constructed based on the observation that the plume is a quasi-neutral, unmagnetized plasma in which the electrons are effectively collisionless. The DSMC method is used to simulate collisional phenomena and an empirical source model is used to simulate an SPT-100 thruster. The model used is based on experimental data and contains no free parameters. The plume simulation was written in ANSI C and runs were carried out on UNIX workstations. Execution times were typically 2-4 hours on axisymmetric geometries.

Results and Discussion

A series of changes have been made to the PIC-DSMC plume model in order to improve its agreement with experimental data. The effects of these changes can be seen by comparing previous results with those produced by the new plume model. Figures 2-4 show axisymmetric simulations of an SPT-100 operating in a vacuum tank with a pressure of 2.2 x 10^{-6} Torr. Figure 2 shows the simulation geometry, figure 3 shows results from the original simulation, and figure 4 shows results from the present plume model. In both cases, the primary beam is visible as a region of high potential near the centerline. In figure 3, a bump is visible in the potential to the front and side of the thruster. This bump is caused by CEX ions and has been observed experimentally in the plumes of ion thrusters. In figure 4, however, no such bump is visible. Figure 5 shows the experimentally measured ion current density 60 cm from the exit of the thruster.[18] Superimposed on the data are simulated results produced using the original plume model. The simulated results show fair overall agreement but, like Figure 3, include a distinct wing structure which is not present in the data. Figure 6 shows the same experimental results compared to the new plume model. As in previous work, additional current has been added to the simulation to compensate for ambient plasma present in the chamber.[3] The simulated results in Figure 6 show much better agreement with the data. In particular, the wing structure present in Figure 5 is absent from Figure 6. Instead, the ion current density varies monotonically from zero to 100 degrees off axis, just as in the data. The improved performance results from two changes made to the original model. First, the new model includes double ions directly in the simulation, so it is no longer necessary to increase the flow of single ions to compensate for the double ion current. A smaller amount of current is still added to compensate for the presence of ambient plasma in the chamber. As in previous work, the amount of current is determined by subtracting the thruster's discharge current from the total current measured in the experiment. This value changes with the tank pressure and is shown in Table 4.

Pressure (Torr)	Integrated Current (A)	Compensation Current (A)
2.2 x 10-5	4.69	0.19
5.6 x 10-6	5.26	0.76
2.5 x 10-5	5.97	1.47
6.3 x 10-5	6.15	1.65
Discharge Current	~4.5	N/A

Table 4: Beam Current vs. Pressure

The second change to the original simulation was to modify the axial ion temperature used in the ion source model. Originally, a relatively low temperature of 3.4 eV was used in the source model. This value is consistent with experimental measurements made by Manzella using laser induced fluorescence.[19] The new model uses a temperature of 34 eV, which is consistent with RPA data from Manzella and Absalamov.[7,20] The resulting velocity distribution is much wider with respect to the drift velocity and consequently changes the structure of the plume region. This interaction merits some discussion.

Careful study of the axisymmetric simulations indicates that the presence of the wing structure in Figure 3 is related to the turning radius of ions leaving the anode. As the plume expands, the density gradient at the edge of the plume creates an electric field that turns ions away from the centerline at a rate which is related to their initial velocity. Ions with relatively high velocities tend to follow straight trajectories, while ions with slow velocities are quickly turned towards the edge of the plume. In ion thrusters, the ions in the main beam are moving in a small range of velocities which is much faster than ions created by CEX, so the CEX ions exit to the sides of the plume and form a "wing" like structure (similar to that shown in Figure 3). In a Hall thruster, the beam as a whole is moving much faster than the CEX ions, but the axial ion temperature is relatively high, so the beam ions span a wide range of velocities. As a result, different parts of the beam turn at different rates, producing a much less defined beam boundary. The slower ions at the edge of the beam end up merging with the CEX ion wings to create a relatively smooth density variation. The presence of wings is therefore highly dependent on the axial ion temperature. If the ion temperature is low, the beam is relatively tight, and the CEX wings are clearly visible. If the ion temperature is high, the beam is relatively diffuse, and the CEX wings are absorbed into the diffuse beam structure.

There are several conclusions which can be drawn from this discussion. First, the reported axial ion temperature of 3.4 eV is probably in error and should be examined carefully. Second, the relatively tight plumes associated with ion thrusters are a function of their relatively low axial ion temperature. And third, the key to lowering the SPT's beam divergence may be to lower the axial ion temperature rather than alter the radial ion distribution. One way to do this may be to shorten the anode, though it is not clear what effect this would have on the performance of the thruster.

As with the original plume model, a series of simulations were run to examine the effect the that background pressure present in ground test facilities has on the structure of the SPT-100 plume. Four different ambient pressures were simulated, each corresponding to an experimental result reported by Manzella.[18] The presence of background gas was simulated by adding neutral Xenon particles in a Maxwellian distribution and by including thermal Xenon fluxes along boundaries with the ambient plasma (as shown in Figure 2). The background ion temperature was assumed to be 300 K and the background number density was determined using the ideal gas law. Typical results are shown in Figure 7.

Figure 7 shows the simulated ion current density as measured at four different ambient pressures in an arc 60 cm from the thruster exit. For comparison, experimental measurements by Manzella are shown in figure 8. The simulated results show good agreement with experimental data, and the same shapes and trends are present in both sets of results. In addition, the results agree fairly well numerically, within a factor of 2 to 3 across the domain, with larger disagreement at very high and very low angles. The simulated results do tend to overpredict the current along the centerline and underpredict it at high angles. The overprediction along the centerline may be a result of numerical artifacts along the centerline, inaccuracies in the source model, or saturation of the ion current probe during the experiments. The underprediction at high angles may be a result of errors in measured CEX cross sections (which is estimated at 20%)[12] or inaccuracies in the source model. We note also that the errors could be present in the experimental measurements, particularly in the pressure measurements which are thought to be accurate to within only half a decade.[21] The error on the ion current measurements is unknown, but measurements of this type are typically accurate to within about 50%.[22] In view of the uncertainty, it can be said that the PIC-DSMC model agrees well with experimental data, even as the ambient pressure is varied by an order of magnitude. This verifies the accuracy of the underlying PIC-DSMC algorithm.

Erosion Rates

In order to verify the surface interaction model, the results from the PIC-DSMC model were compared with experimentally measured erosion results from Randolph et al.[16] Randolph placed samples of silver foil, solar cell interconnectors, and solar cell coverglasses in a vacuum tank and exposed them to the SPT-100 plume for a period of 200 hours. These tests were conducted at a pressure of 3×10^{-5} Torr and collimators were used to protect the samples from contamination. Figure 9 shows simulated results for the erosion of Silver after two hundred hours of exposure at a pressure of 2.5×10^{-5} Torr. The results of Randolph and Pencil are superimposed on the simulated results. In both cases, the measurements are taken 1 m from the anode exit. Although the simulated results are noisy, they agree fairly well with the experimental data. The same shapes and trends and clearly present in both results, though the simulation consistently underpredicts the measured erosion, sometimes by more than a factor of 2. Figure 10 shows similar results for the erosion of Quartz after 200 hours of exposure to the plume. Superimposed on these results is the experimentally measured erosion of solar cell coverglasses after 200 hours of exposure to the plume. Although the same trends are apparent, the agreement is not as good, and the simulation consistently underpredicts the data by a factor of three or more. The relatively high quality of the results shown in Figure 9 suggests that the source of the inaccuracy is probably a problem with the relationship used for the sputtering coefficient. The results suggest that the material used in the solar cell coverglasses has higher sputtering coefficient than the "quartz glass" modeled in the simulation. In addition, the sputtering coefficients used in the simulation are valid for ions striking normal to the surface. In reality, however, ions which approach at high angles may have sputtering coefficients several times higher than those used in the simulation.[23] Unfortunately, no data is available on the angular dependence of the sputtering coefficient of Xenon atoms on silver or Quartz surfaces. Such data is necessary to further refine the sputtering model.

RPA Measurements

A retarding potential analyzer (RPA) has also been used to measure the energy of ions in the plume. Because the PIC-DSMC algorithm is a particle based method, an RPA can be simulated by recording the energy of macroparticles crossing imaginary surface. Figure 11 shows a simulated RPA distribution at a chamber pressure of 6.3×10^{-5} Torr. Figure 11 was constructed by recording the energy of particles crossing an arc 60 cm from the anode exit. Particle energies were recorded in increments of 0.5 degrees, so the curve marked "7.5 degrees" corresponds to ions crossing at angles between 7.25 and 7.75 degrees from the centerline. Figure 11 shows experimental RPA data taken along an arc 1 m away from the thruster at a pressure of 5 to 6 x 10^{-5} Torr.[20] Although there is some agreement between simulated and real RPA data, the overall level of agreement is not so good. There are two "peaks" present in both sets of data. The first is a low energy peak near the origin that represents slow moving CEX ions. The second is a high energy peak that represents beam ions coming directly from the anode. In the simulated data, the low energy peak is narrow and located right at the origin. In the experimental data, the low energy peak is slightly wider and is offset from the origin by about 20 V. This seems to indicate that the RPA itself is floating at some potential relative to the plasma, which would effect the accuracy of the experimental results. The high energy peaks are offset from each other by almost 75 eV. The peak of the experimental data is at about 275 eV, while the peak in the simulated data is just over 200 eV. The placements of the experimental peak is interesting because an average ion energy of 275 eV would correspond to an Isp of 2048 sec, which is considerably higher than experimentally observed Isp of 1735 sec. (excluding cathode flow). Subtracting 25 eV from the peak gives an Isp of 1971 sec., which is still much higher than the experimentally observed value. This disagreement suggests that the RPA may be substantially biased with respect to the ambient plasma. Further work is needed to determine the source of these inconsistencies.

Figures 11 and 12 differ qualitatively as well as quantitatively. In particular, the high energy peak is much broader in the simulated data and the location of the peak varies with the RPA's angle from the centerline. This behavior is consistent with the earlier observation that particles with low energies are more easily turned by the radial electric fields present in the plume. The experimental data, however, shows no such variation. In fact, although the magnitude of the peak drops rapidly, the location of the peak is the same at all distances from the centerline. Why this occurs is unclear. The implication may be that the SPT-100 is emitting high energy ions in directions pointing well away from the centerline. These ions could mask the energy shifting effect caused by the plume expansion. An interesting characteristic of the RPA data is the presence of a high energy "tail" of ions with energies hundreds of volts above the 300 V cathode-anode drop. Tails of this type are sometimes produced by plasma instabilities, so the presence of a tail may indicate that instabilities play an important role in the creation of high energy ions moving at large angles relative to the centerline. More work is needed to understand the reasons for these disagreements and the role instabilities play in the SPT-100 thruster.

In summary, the PIC-DSMC model presented in this paper has been compared to a wide range of experimental data. The model shows good agreement with ion current density measurement in the plume region, but compares less well to RPA data. The reason for these disagreements is unclear, as some inconsistencies seem to be present in the experimental data. In addition, a sputter erosion model has also been described and compared to experimental data. This model shows good agreement with measured erosion rates for silver interconnectors, but is less accurate when predicting the erosion of solar cell coverglasses. More accurate measurements of the coverglass sputtering coefficient are needed to improve these results. The results shown in Figure 8 also demonstrate that the ambient neutrals present in ground based test have a significant effect on the backflow currents caused by CEX collisions. This implies that ground based tests will overpredict the degradation which will occur on surfaces on orbiting spacecraft. To study these issues, axisymmetric simulations of a thruster operating in vacuum and three dimensional simulations of realistic spacecraft configurations are presented in a companion paper.

Conclusions

A computational model of an SPT plasma plume has been constructed using a quasi-neutral PIC-DSMC model. This model is based on theoretical work showing that the plume consists of a quasi-neutral plasma with collisionless electrons in which the magnetic field can be neglected. The resulting simulation can accurately model an SPT-100 plume on meter length scales with run times of 4-6 hours. Comparisons between the computational model and a variety of experimental data indicate that the PIC-DSMC simulation can accurately predict important features in the plume region, including the ion current density and the erosion rate for materials exposed to the plume. However, comparison with RPA data shows the limitations of the present model and highlights several questions about the SPT-100 thruster. The RPA data indicates that high energy ions are present throughout the plume region, even in regions far from the centerline. The data also indicates that a tail of high energy ions is present in the plume, which may indicate that plasma instabilities play an important role in the creation of high energy ions. Although further work is needed to clarify the role instabilities play in the plume region, the experimental work presented in this paper verify the fundamental accuracy of the PIC-DSMC algorithm. The plume model should therefore prove to be a useful tool for investigating the impact of Hall thrusters on commercial spacecraft, particularly when implemented in a three dimensional geometry.

The PIC-DSMC model offers designers the ability to model Hall thrusters under orbital conditions. The axisymmetric model has been used to study a generic thruster operating in vacuum and the three dimensional model has been used to study realistic spacecraft geometries. These results are discussed in a companion paper. Future work is planned to improve the existing plume model. In particular, the development of an unstructured PIC-DSMC simulation would allow designers to simulate very realistic spacecraft geometries. In addition, new sputtering data will be incorporated into the surface interaction model as it becomes available, and efforts will be made to improve the source model so it gives better agreement with existing experimental data.

Acknowledgments

Thanks to David Manzella at NYMS Inc. for providing us with data used in this paper. Thanks also to Tom Randolph at Space Systems Loral and Prof. Manuel Martinez-Sanchez at MIT for their insight and help. This paper was funded by the Air Force Office of Scientific Research.

References

[1] Rhee, M.S. and Lewis, M.J. "Numerical Simulation of Stationary Plasma Thruster Exhaust Plume," AIAA Paper 95-2928, July 1995.

[2] Dickens, J., Kirstiansen, M. and O'Hair, E. "Plume Model of Hall Effect Plasma Thrusters with Temporal Consideration," IEPC Paper 95-171, September 1995.

[3] Oh, D. and Hastings, D. "Axisymmetric PIC-DSMC Simulations of SPT Plumes," IEPC Paper 95-160, September 1995.

[4] Oh, D. and Hastings, D. "3-D PIC-DSMC Simulations of Hall Thruster Plumes and Analysis for Realistic Spacecraft Configurations," AIAA Paper 96-3299, July 1996.

[5] Manzella, D.H. "Stationary Plasma Thruster Plume Emissions," IEPC Paper 93-097, September 1993.

[6] Arkhipov, B.A. and Kozubsky, K.N. "The Development of the Cathodes-Compensators for Stationary Plasma Thrusters in USSR," IEPC Paper 91-023, October 1991.

[7] Myers, R.M. and Manzella, D.H. "Stationary Plasma Thruster Plume Characteristics," IEPC Paper 93-096, September 1993.

[8] Manzella, David. "Hall Thruster Beam Characterization," AIAA Paper 95-2927, July 1995.

[9] Ruytan, W. M. "Density-Conserving Space Factors for Particle Simulations in Cylindrical and Spherical Coordinates," *J. Comp. Phys.*, Vol. 105, 1993, pp. 224-232.

[10] Elgin, J.B. "Getting the Good Bounce: Techniques for Efficient Monte Carlo Analysis of Complex Reacting Flows," Technical Report No. SSI-TR-28. Spectral Sciences, Inc. Burlington, MA.

[11] Rapp, D. and Francis, W.E. "Charge Exchange Between Gaseous Ions and Atoms," *Journal of Chemical Physics*, Vol. 37, No. 11, 1962, pp. 2631-2645.

[12] Hasted, J.B. and Hussain, M. "Electron Capture by Multiply Charged Ions," *Proc. Phys. Soc.*, Vol. 83, 1964, pp. 911-924.

[13] Koura, K. and Matsumoto, H. "Variable Soft Sphere Molecular Model for Air Species," Phys. Fluids A, Vol. 4, No. 5, May 1992, pp. 1083-1085.

[14] Samanta Roy, R.I. "Numerical Simulation of Ion Thruster Plume Backflow for Spacecraft Contamination Assessment," Ph.D. Thesis, MIT Department of Aeronautics and Astronautics, June 1995, pg. 81.

[15] Dalgarno, A., McDowell, M.R.C. and Williams, A. "The Mobilities of Ions in Unlike Gases," *Philosophical Transactions of the Royal Society of London*, Vol. 250A, No. 982, 1958, pp. 51-425.

[16] Randolph, T., Pencil, E. and Manzella, D. "Far-Field Plume Contamination and Sputtering of the Stationary Plasma Thruster," AIAA Paper 94-2855.

[17] Rosenberg, D. and Wehner, G.K. "Sputtering Yields for Low Energy He+-Kr+-, and Xe+-Ion Bombardment," Journal of Applied Physics, Vol. 33, No. 5, May 1962, pp. 1842-1845.

[18] Manzella, D.H. and Sankovic, J.M. "Hall Thruster Ion Beam Characterization," AIAA Paper 95-2927, July 1995.

[19] Manzella, D. "Stationary Plasma Thruster Ion Velocity Distribution," AIAA Paper 94-3141, June 1994.

[20] Absalamov, S.K. et al. "Measurement of Plasma Parameters in the Stationary Plasma Thruster (SPT-100) Plume and Its Effect on Spacecraft Components," AIAA Paper 92-3156, July 1992.

[21] Manzella, David. Personal Communication, 1996.

[22] Gallimore, Alec. Personal Communications, 1996.

[23] Chapman, Brian. *Glow Discharge Processes: Sputtering and Plasma Etching*. Wiley and Sons, New York, 1980. pg. 247.

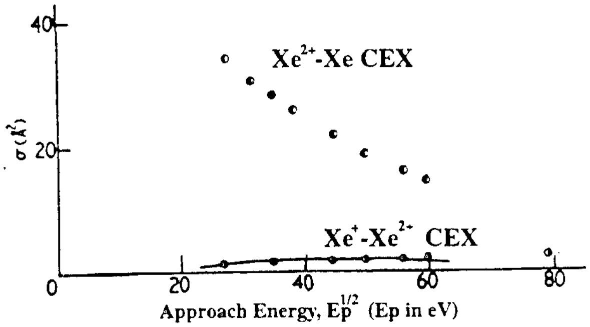


Figure 1: Xenon Double Ion Charge Exchange Cross Sections

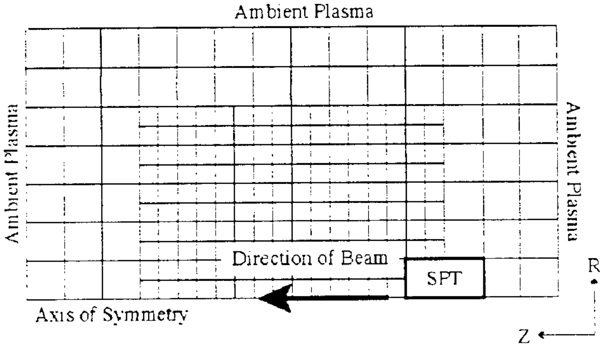


Figure 2: Grid and Domain

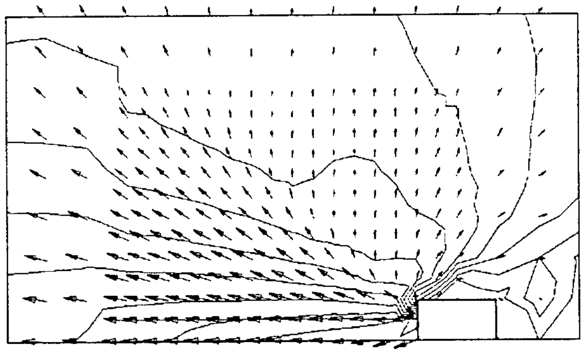


Figure 3: Contour Plot of Potential and Vector Plot of Current Density (Original Model)

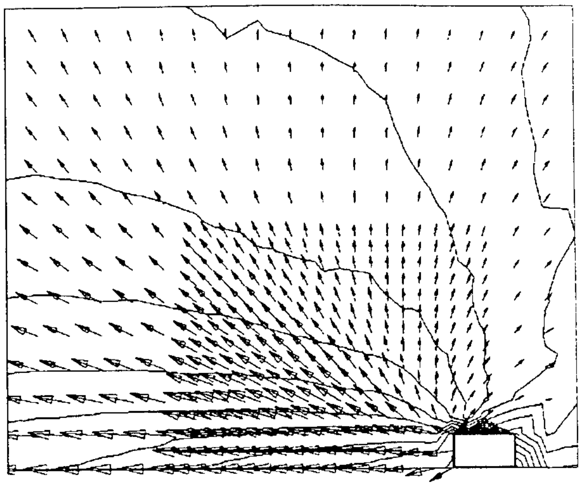


Figure 4: Contour Plot of Potential and Vector Plot of Current Density (Improved Model)

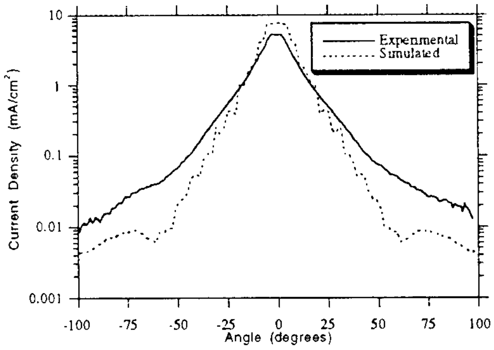


Figure 5: Comparison of Ion Current Density Data with Original Model ($z = 60$ cm, $P = 6.3 \times 10^{-6}$ Torr)

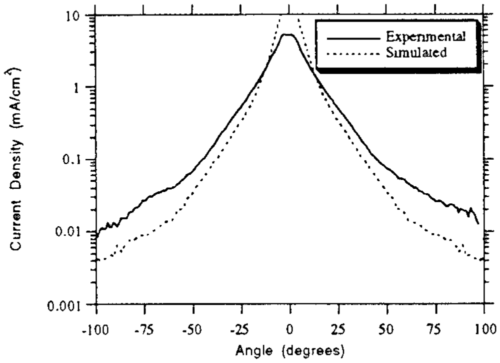


Figure 6: Comparison of Ion Current Density Data with Improved Model ($z = 60$ cm, $P = 6.3 \times 10^{-6}$ Torr)

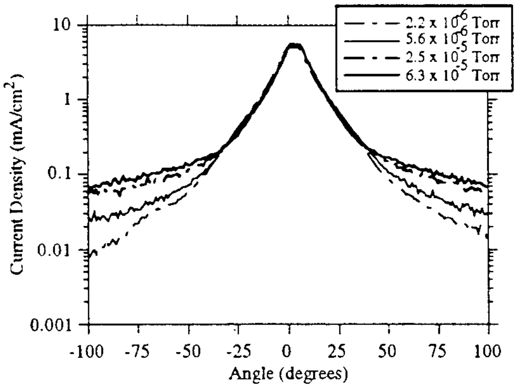


Figure 7: Measurements of Ion Current Density as a Function of Facility Pressure (z = 60 cm)

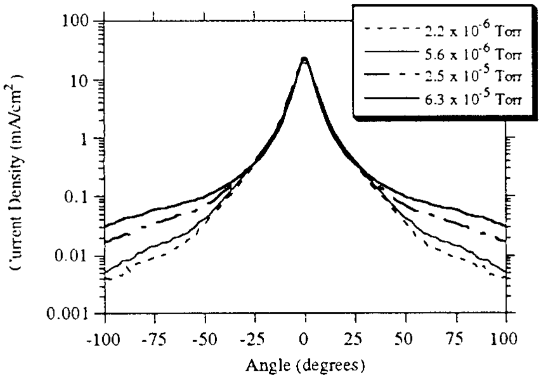


Figure 8: Simulated Ion Current Density as a Function of Ambient Pressure (z = 60 cm)

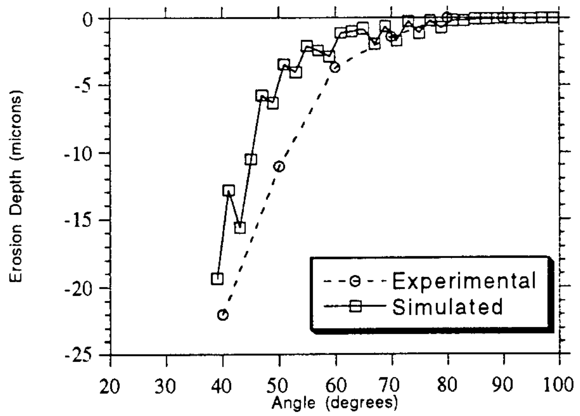


Figure 9: Comparison of Calculated Silver Erosion Rates to Experimental Data (z = 1 m)

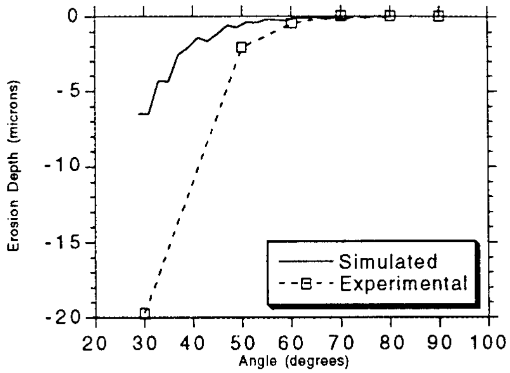


Figure 10: Comparison of Calculated Coverglass Erosion Rates to Experimental Data (z = 1 m)

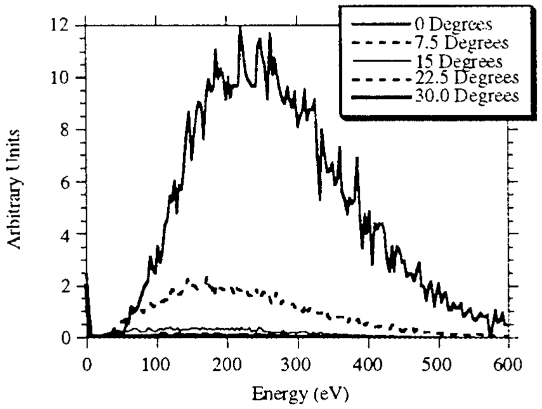


Figure 11: Simulated RPA Data Showing Ion Energy Distribution (z = 60 cm)

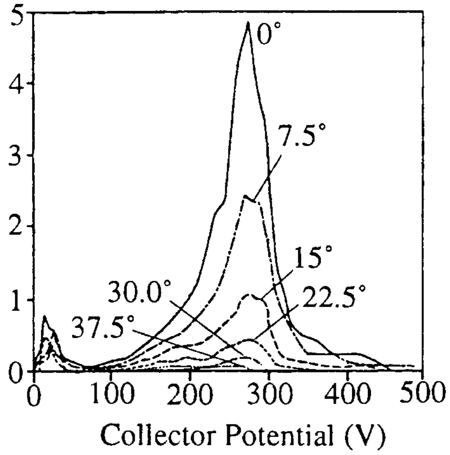


Figure 12: Experimental RPA Measurements Showing Ion Energy Distribution (z = 1 m)

BASIC RESEARCH HELPS PUT HALL THRUSTERS ON NEXT GENERATION COMMUNICATION SATELLITES

Alec Gallimore, Daniel Hastings, and Brian Gilchrist

It currently costs more than $12,000 to send one pound to orbit, which means that a satellite on station is worth at least twice its weight in pure gold. So it is critical that satellite designers save weight in order to save money. There are two approaches to achieving this goal: 1) make the spacecraft smaller and lighter by using micro- or even nano-satellite technology; or 2) reduce the spacecraft propellant mass by using fuel-efficient propulsion systems.

Electric propulsion can reduce dramatically the amount of fuel that is needed by satellites. The basic principle is simple: instead of using chemical energy to drive fuel out of a nozzle, electrical energy is used to drive the fuel much faster, giving the satellite a greater "kick" for each pound of fuel. A variety of different devices have been constructed on this principle, most notably the Hall thruster. But to drive the flow, the fuel must be transformed into a charged gas known as a plasma, the most abundant form of matter in the universe.

A plasma is a gas that consists not only of neutral atoms, but also of freely-moving charged particles. An electric thruster releases a plume of very fast-moving charged particles which form a large cloud around a satellite. While this may sound harmless, when a particle moving at 40,000 miles per hour hits a solar array, it can damage the cells and interconnectors which provide critical power to the satellite. In fact, one of the main reasons that designers have not employed electric thrusters is their fear that the thrusters will damage these arrays, leaving the satellite stranded thousands of miles from Earth. Another concern satellite manufacturers have is the plume's possible effects on incoming and outgoing radio signals. Since charged particles interact with electromagnetic signals, radio signals which propagate through this plume may become distorted. This means that a television picture could appear fuzzy or not at all if its carrier signal were to propagate through the thruster exhaust.

Until recently, designers had no way to figure out where these charged particles would go, what damage they would cause, or their influence on radio signals. However, electric propulsion researchers who specialize in theoretical and applied plasma physics have teamed up to tackle this problem both computationally and experimentally.

In support of this effort, a three dimensional computer model of an electric Hall thruster and its plasma plume was developed [1-3]. Over a few hours, the program not only calculates where the plume will go, but what the plume will do to the surfaces it touches. It will even calculate the amount of erosion a solar cell coverglass or interconnector will experience over the lifetime of the satellite. If the damage is unacceptable, the designer can simply move the thruster and try again until an acceptable design is obtained.

In Figure 1, a plasma plume from a Hall thruster is shown. In Figure 2, the angle between the thruster exhaust plume axis and the solar array (cant angle) is shown as the array turns. Any array erosion greater than 1.2 microns a month is unacceptable. Thus, this design map shows that satellite manufacturers must cant their thrusters at angles of 40 degrees relative to the ideal thrust axis. This code has been used by spacecraft systems engineers at Hughes and at Lockheed-Martin to determine the best location for Hall thrusters on their new HS 702 and A2100 communication satellites, respectively.

Experiments presented in References [4-5] provided much of the experimental plume plasma data that was needed to calibrate and validate the code. A state-of-the-art electrostatic ion energy analyzer was developed to measure the ion kinetic energy distribution in the plume (see Figure 3). In addition, heat flux sensors and a neutral particle flux probe - a new invention of ours - were used to characterize the plasma and neutral particle transport properties of the plume. This was the first time both the charged and neutral particles of a Hall thruster plume were measured simultaneously. Data from these experiments are being used at the Space Systems/Loral Corporation to integrate Hall thrusters with their new Lafayette communication satellite.

In addition, L- (1.6 GHz) and Ku-band (17 GHz) signals were transmitted through the exhaust plume of the Hall thruster, received, and analyzed at various locations to determine how radio waves are affected by the charged particles in the plume [6-7]. Figure 4 shows the effect the plume plasma has on a Ku-band radio signal propagating through the center of the plume 0.15 m from the thruster exit. In addition to increased signal noise, coherent sideband peaks are seen at discrete frequencies (multiples of 26 kHz) corresponding to discharge oscillations of the engine. An electromagnetic ray-tracing code was developed to predict these and other effects at varying frequencies and distances from the thruster so that system designers could adequately account for these effects. There is considerable agreement between the code predictions and our experimental results, suggesting that both amplitude and phase effects are important.

References

1. Oh, D. and Hastings, D., “Axisymmetric PIC-DSMC Modeling of SPT Plumes,” IEPC paper 95-160 presented at the 24th International Electric Propulsion Conference, September 1995, Moscow, Russia.

2. Oh, D. and Hastings, D., “Experimental Verification of Axisymmetric and Three Dimensional PIC-DSMC Simulations of Hall Thruster Plumes,” AIAA paper 96-3196 presented at the 32nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, July 1996, Orlando, Florida & submitted to Journal of Propulsion & Power

3. Oh, D. and Hastings, D., “3-D PIC-DSMC Simulations of Hall Thruster Plumes and Analysis for Realistic Spacecraft Configurations,” AIAA paper 96-3299 presented at the 32nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, July 1996, Orlando, Florida & submitted to Journal of Propulsion & Power

4. King, L. B., and Gallimore, A. D., "A Gridded Retarding Pressure Sensor for Ion and Neutral Particle Analysis in Flowing Plasmas," Review of Scientific Instruments, Vol. 68, No. 2, Feb., 1997, 1183-1188.

5. King, L. B., and Gallimore, A. D., "Ionic and Neutral Particle Transport Property Measurements in the Plume of an SPT-100," (AIAA-96-2712): 32nd Joint Propulsion Conference, Orlando, FL, July, 1996 (submitted to AIAA Journal of Propulsion and Power).

6. Ohler, S. G., Gilchrist, B. E., and Gallimore, A. D., “Electromagnetic Signal Modification in a Localized High Speed Plasma Flow: Simulations and Experimental Validation of a Stationary Plasma Thruster (SPT),” IEEE Transactions on Plasma Science, 1996.

7. Gilchrist, B. E., Ohler, S. G., and Gallimore, A. D., "Flexible Microwave System to Measure the Electron Number Density and Quantify the Communications Impact of Electric Thruster Plasma Plumes," Review of Scientific Instruments (AIP), Rev. Sci. Inst., Vol. 68, No. 2, Feb., 1997, 1189-1194.

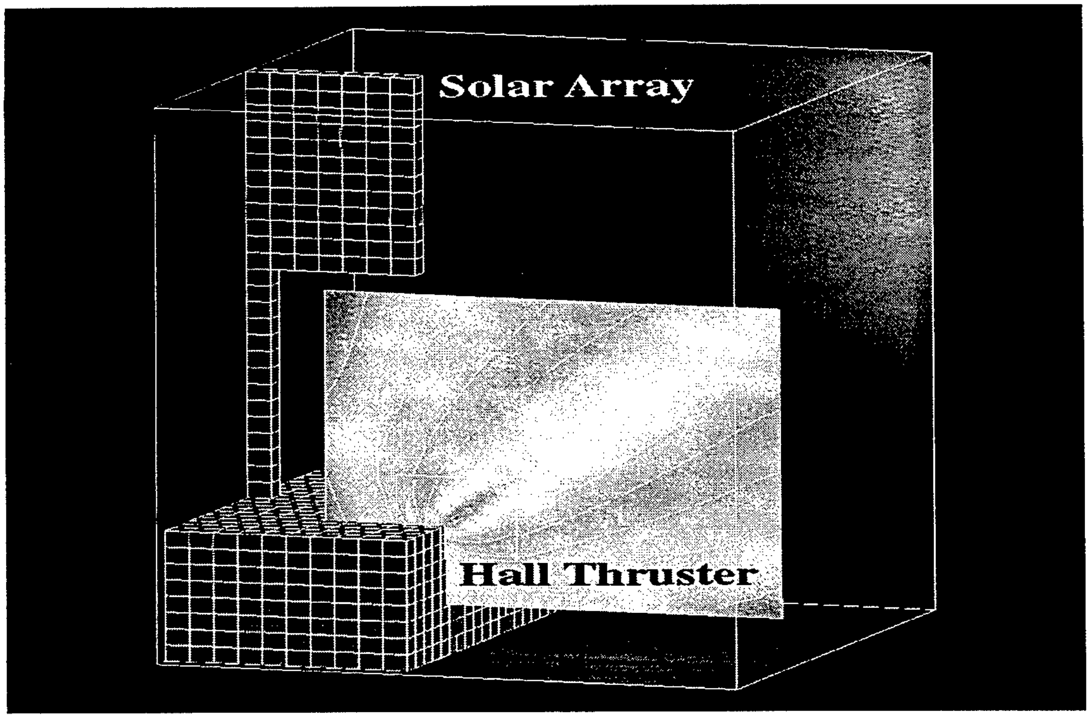


Figure 1: Predicted Hall thruster plume expansion profile from the computer model. By using this model, designers can see how a plasma plume affects the entire satellite.

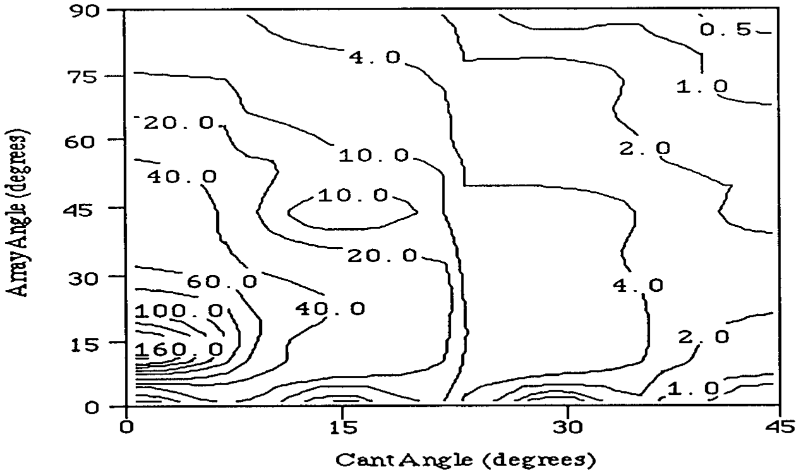


Figure 2: Erosion Rate for Silver at Corner of Array for an SPT-100 Thruster operating at a xenon flow rate of 5.37 mg/sec., an I_{sp} of 1610 sec. The solar array yoke length is 1.9 m and the array width is 1.14 m. Erosion rates are in microns/month.

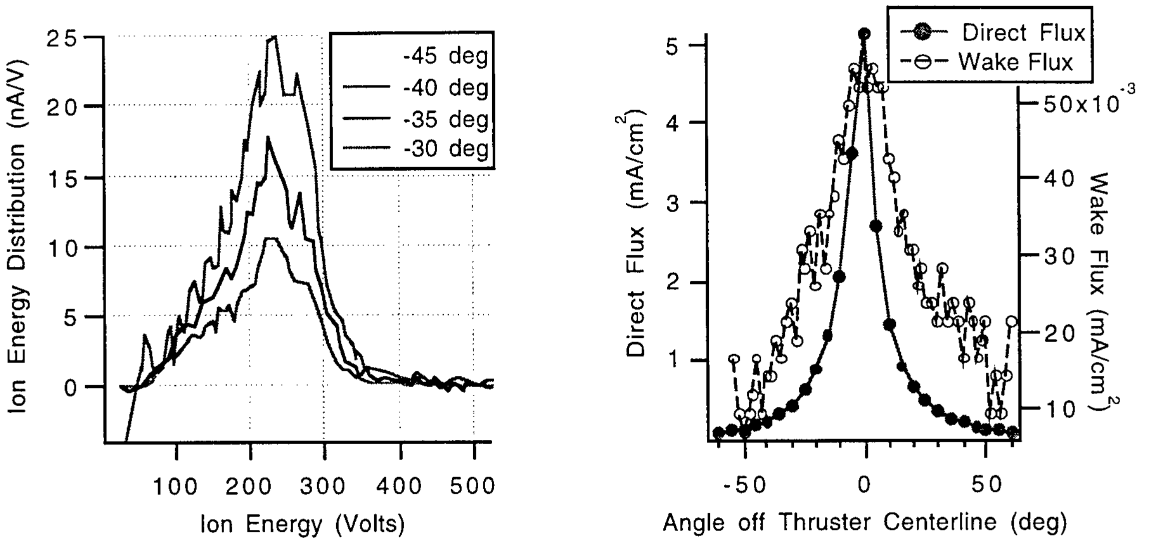


Figure 3: Ion energy and ion beam current density distribution in the plume of an SPT-100 Hall thruster taken at a radius of 0.5 m from the thruster exit plane, from centerline to over 90 degrees with respect to the thruster axis. These data were collected with a four-grid retarding potential energy analyzer (RPA). A Faraday probe provided ion current density measurements. To correct for stray charged particles in the vacuum chamber, a second Faraday probe was mounted in the wake of the thruster-facing electrode to measure the wake ion flux due to facility effects. The wake current was found to be negligible.

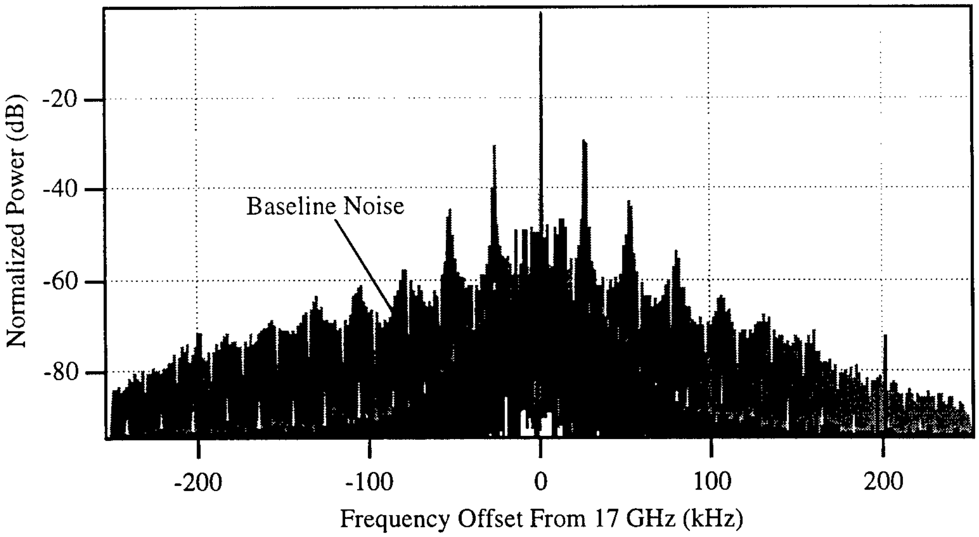


Figure 4: Power spectral density for a 17 GHz signal transmitted through the center of thruster plume 0.15 m downstream.

About the Authors

- Alec Gallimore is an Assistant Professor of Aerospace Engineering and of Applied-Physics at the University of Michigan. He teaches and conducts experimental research on chemical and advanced space propulsion, and in basic and space plasma physics. He is a member of the International Electric Propulsion Technical Committee and the DARPA-sponsored Defense Science Study Group.

- Daniel Hastings is a Professor of Aeronautics and Astronautics at the Massachusetts Institute of Technology. He teaches and conducts research on advanced space propulsion, space systems, and space environment effects on space systems. He is a member of the Air Force Scientific Advisory Board and the National Academies Aeronautics and Space EngineeringBoard.

- Brian Gilchrist is an Associate Professor of Electrical Engineering at the University of Michigan who specializes in experimental plasma electrodynamics and the development of active and passive sensor systems for space plasma applications. He is the Principal Investigator for the Shuttle Electrodynamic Tether System (SETS) experiment flown on the STS-75 shuttle mission in 1996 as part of the Tethered Satellite System (TSS) mission. Prior to receiving his Ph.D., he held industrial R&D and management positions for ten years.